THE UNIVERSITY OF ALBERTA

TWO-DIMENSIONAL HARMONIC ANALYSIS OF POTENTIAL FIELDS

BY

© RAM GOPAL AGARWAL

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF PHYSICS

EDMONTON, ALBERTA

JUNE, 1968

Thesis
1968 (F)
1 D

UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies for acceptance, a thesis entitled TWO-DIMENSIONAL HARMONIC ANALYSIS OF POTENTIAL FIELDS, submitted by Ram Gopal Agarwal, in partial fulfilment of the requirements for the degree of Doctor of Philosophy.

ABSTRACT

In this investigation, techniques such as cross-correlation, convolution or filtering, and spectral density estimates have been developed in conjunction with a two-dimensional fast FOURIER transform algorithm in the digital domain to allow a more complete examination of any set of potential field data. In addition a new and novel method of cross-correlation has been formulated and has been applied to determine automatically the trends on any geological or geophysical map. This is accomplished by digital computation and introduces a minimum amount of human bias. Thus it is possible to distinguish and weight trends due to various geological processes and separate them by a two-dimensional filter. The physical properties of the rocks may then be analysed more completely from the residuals of the potential field.

In another part of this investigation, programs to carry out the upward and downward continuation, derivatives and reduction of the total magnetic field to the pole, have been developed using two-dimensional fast FOURIER transform algorithm. The techniques have been applied to the magnetic and gravitational field data to determine a $\frac{J}{\rho}$ ratio for various rock units, which in addition provides information on the direction of magnetization of the source body. A

coherency test is used to evaluate the validity of the determination of $\frac{J}{\rho}$ as a function of wave numbers. An example of gravity and magnetic data from Stony Rapids, Northern Saskatchewan illustrates the techniques.

ACKNOWLEDGMENTS

I wish to express my sincere thanks and gratitude to Dr. E.R. Kanasewich for supervising and suggesting the research problem. I am also grateful for his continuous encouragement and invaluable advice at various stages of this project.

My thanks are due to Dr. M. Innes, Gravity Division, Dominion Observatory, Ottawa, for his initiative in undertaking and financing the Gravity Survey in the study area. Mr. R.K. McConnell took great pains in organizing and supervising the gravity field party, and processing of the field data.

I would like to thank Mrs. W.I. Gough and Mr. C.M. McCloughan for many hours of assistance in processing the data and preparing the final diagrams. Mrs. M. Lybacki is commended for her careful and excellent job in typing.

I am particularly thankful to Mr. W.J. Wattamaniuk for discussions and assistance in computing and Mr. V. De La Cruz for assistance with some of the mathematical aspects of this topic. Thanks are also due to Mr. G. Pratt, scientific applications group, for discussions on computing problems.

During the course of this study, the financial support by the Department of Mineral Resources, Government of Saskatchewan is greatly appreciated. The study was also supported by the grants from the National Research Council

of Canada.

Finally, I wish to thank my wife for her patience and understanding during the long years of graduate studies.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

CHAPTER I

INTRODUCTION

Regional studies of structural and tectonic features have been in progress for a number of years. Information on large scale tectonic trends in the continental areas may be derived by a wide variety of geological methods or by comprehensive geophysical surveys of the gravitational and magnetic fields. Analytic methods for determining the nature of tectonic and structural features are often limited and are more subject to human bias than is desirable. An attempt is made in this thesis to develop a novel technique for trend analysis by digital computation on a two dimensional time series which will reduce this bias. The analysis makes use of auto-correlation and cross-correlation functions, the power spectral density, convolution or filtering and such powerful algorithms as the fast Fourier transforms. It is possible to envisage one set of structural trends developing in response to a certain field of stress during one geological period and a different trend forming at some later stage in the development of the crustal section. It is desirable and may be possible to separate such independent features, on the basis of their physical properties, through a statistical study in the spatial domain.

The application of harmonic analysis rests on a well developed theoretical foundation and has been applied in a number of interesting cases. Tsuboi and Fuchida (1938)

have used such techniques in the interpretation of gravity anomalies to estimate the corresponding subterranean mass distribution. Tsuboi (1959) represented discretely sampled gravity field data, at any elevation z, by a double Fourier series expression:

$$\Delta g(x,y,z) = \sum_{m=0}^{m_0} \sum_{n=0}^{n_0} B_{mn}\, e^{\frac{2\pi z}{L}\sqrt{m^2+n^2}} \begin{matrix}\cos\\ \sin\end{matrix}\, m\frac{2\pi x}{L} \begin{matrix}\cos\\ \sin\end{matrix}\, n\frac{2\pi y}{L} \qquad \text{.......(1.1)}$$

where L is the wavelength in the x and y directions.

B_{mn} are the coefficients in the Fourier series expansion and may be used to determine the amplitude spectrum.

m_0, n_0 are the highest harmonics along the x and y axis as determined by the digital interval.

$\Delta g(x,y,0)$ is the Bouguer anomaly on a plane surface of the earth and is expressed in terms of a Fourier series by equation (1.1) for $z = 0$. One can project the field above the earth for z taken as negative and downward closer to the source if z is considered positive as desired in the upward and downward continuation operations. The general details of harmonic analysis and its application to magnetic fields

has been discussed by various authors (Chapman and Bartels 1940; Vestine and Davids 1945). Bullard and Cooper (1948) employed similar analytic methods in the study of downward continuation of gravitational fields. Swartz (1954) has used harmonic analysis to represent the gravity and magnetic data in the wavelength domain and prepared residual maps from theoretical contour maps by a two dimensional convolution using somewhat arbitrarily calculated coefficients. The weighting was carried out on various grid systems or template patterns and the filter operation was evaluated by studying the attenuation curves which show the amplitude for various sizes and shapes of original anomalies.

A study by Alldredge et al (1961, 63) indicated that terrestrial magnetic anomalies are caused by two types of sources; for wavelengths shorter than 300 km. he attributed the field to sources in the crustal layers; for wavelengths longer than 3000 km. the sources are apparently at the core-mantle boundary. Their power density study indicated that there was very little energy for intermediate wavelengths. Other aspects relating to model calculations, spectral representation, curie point geotherm etc. are discussed by Serson and Hannaford (1957), Dean (1958), Gladkii (1960), Goldstein (1962), McClure (1963), Byerly (1965), Meskó (1965), Odegard and Berg (1965), Hahn (1965), Bhattacharyya and Morley (1965), Fuller (1966), Gudmundsson (1966),

Spector and Bhattacharyya (1966). Bhattacharyya (1966) has investigated the two-dimensional spectrum analysis of the total magnetic field anomaly due to a rectangular prismatic body with an arbitrary direction of magnetization. From this study, he has concluded that there is a shift of the spectrum towards the longer wavelength end, with increase in either depth or horizontal dimension, or in both, of the magnetized body. Thus by removing the short wavelengths it is possible to study the effects due to long wavelength sources only. One can also estimate the dimensions of the body by noting the wavelengths at which the amplitude spectrum becomes negligible. There is, as always with the potential methods, an ambiguity in the interpretation and the peaks and widths of the amplitude spectrum are also indicative of the strength of the source, depth and vertical extent of the magnetized body.

Various attempts have been made to obtain the significant trends of mapped data. Affleck (1963) carried out statistical study on selected portions of magnetic data for trend determination. In this study he emperically determined and measured the length of each positive or negative magnetic anomaly axis. The trend was defined as the angle measured clockwise from geodetic north to these axes. He grouped the anomaly trends in 5-degree catagories with the length of all trends within a catagory being summed and the results

were presented in the form of a trend rosette. Affleck concluded that "the dominant trend is commonly NW-SE or NE-SW" and his conclusions are mainly valid for the North American Continent as most of the data available was from Canada and the U.S.A. However his method for trend determination and anomaly spacing was somewhat arbitrary and subject to individual bias. Hall (1964) carried out a trend analysis on Texada Island, B.C., Canada, in which he tried to reduce the subjectivity in his analysis by dividing the area into squares of one mile on a side and defining these squares as the "elements" of the pattern. It was assumed that any linear feature that does not pass out of a square is counted as one element and any feature which passes through several squares is counted as having that number of elements. He measured the direction of each element as an azimuth clockwise from north and then the number of elements in each 5-degree of azimuth were counted. The results were presented in the form of frequency curves. Various profiles were also analyzed by means of a one dimensional auto-correlation and cross-correlation function.

Horton et al (1964) attempted to determine trends with a two dimensional auto-correlation function:

$$A(r,s) = \frac{\sum_{i=1}^{N-r} \sum_{j=1}^{N-s} X_{i,j} \; X_{i+r,j+s}}{\sum_{i=1}^{N} \sum_{j=1}^{N} X_{i,j}^{2}} \qquad \ldots\ldots (1.2)$$

where $r,s = 0, \pm 1, \pm 2, \ldots\ldots\ldots, \pm m$

m = maximum lag or displacement

$X_{i,j}$ = is a set of sample data

N = Total number of data points.

Unfortunately, the auto-correlation function tends to provide an ambiguous picture about the trends since the properties of the function are such that one needs to make only half the calculations. The other half may be obtained by using the inherent symmetric properties of auto-correlation, i.e.

$$A(r,s) = A(-r,-s) \qquad \ldots\ldots (1.3)$$

The dominant trends obtained in their analysis are apparent although somewhat accentuated. However, the secondary trends obtained have no basis because they have been derived by joining the negative or positive values in various quadrant corners and these are merely generated by the symmetric properties of the auto-correlation.

From another aspect of the potential field study, we know that the gravity and magnetic fields are related by Poisson's equation:

$$V = -\frac{J}{G\rho} \frac{\partial}{\partial \nu} U \qquad \text{...... (1.4)}$$

where

U is the gravitational potential due to the mass of a body with uniform density ρ.

G is the gravitational constant.

V is the magnetic potential due to the same body polarized uniformly in a direction $\vec{\nu}$ with an intensity of magnetization J.

By differentiating one can obtain the vertical magnetic field anomaly Z_T:

$$Z_T = \frac{J}{G\rho} \frac{\partial}{\partial \nu} g = \frac{J}{G\rho} g'(\nu) \qquad \text{...... (1.5)}$$

Where $g'(\nu)$ is the gravity field gradient in the direction of magnetization. By knowing the inclination I and declination D of the direction of magnetization, one can express the relation as:

$$Z_T = \frac{J}{G\rho} \{g'(z) \sin I + g'(H) \cos I\} \qquad \text{...... (1.6)}$$

where

$g'(z)$ is the first vertical derivative of gravity.

$g'(H)$ is the first horizontal derivative of gravity along the declination D.

We notice that in the above expression one needs to evaluate the first vertical and horizontal gravity derivatives. Special cases such as second derivatives have been dealt by Swartz (1954) and Meskó (1965) using Fourier methods, whereas Bhattacharyya (1965) has discussed the general method of obtaining any derivatives by using Fourier series representation. According to his method one can evaluate the vertical derivatives of the r^{th} order simply by multiplying the coefficients B_{mn} in equation (1.1) with:

$$\left\{ \frac{2\pi}{L} (m^2+n^2)^{\frac{1}{2}} \right\}^r \qquad \ldots\ldots (1.7)$$

From the historical point of view, Eötvös used Poisson's equations to obtain the relations between the three components of the magnetic field with the values recorded on a torsion balance. Garland (1951) applied Poisson's equation to known gravity and magnetic data to obtain the ratio J/ρ for an area in Arkansas, U.S.A. The application of statistical techniques in the digital domain will be explored by this form of analysis in a later chapter.

An important new method of interpreting the magnetic anomalies was developed by Baranov (1957), using Poisson's relations, in which the total magnetic field is transformed to the magnetic pole. Usually a distorted picture of the magnetic anomalies is obtained because the rocks are magnetized in a different direction than the present earth's

magnetic field. Moreover in the case of aeromagnetic surveys the magnetometer may not be aligned exactly in the direction of the total field. Because of these reasons, the apex of the magnetic anomalies is displaced from the underlying sources. The purpose of the transformation is to reduce this distortion and to obtain the magnetic anomaly maps which would result if the whole area was translated to the North Magnetic pole, a region where the inducing field was vertical (i.e. the magnetic pole). In a recent paper, other numerical details have been discussed by Baranov and Naudy (1964). The main drawbacks are that the effects of the remanent magnetization have not been taken into consideration and the method can not be applied for small inclinations of magnetization. Bhattacharyya (1965) has discussed the transformation by using the Fourier series representation and has taken into consideration both induced as well as remanent components of magnetization. Moreover he has obtained the expressions by which one can obtain the magnetic anomalies for any inclination of magnetization and at any magnetic latitude. Magnetic anomalies calculated by Baranov's transformation to the magnetic pole are directly comparable to the quantity on the left hand side of equation (1.6), making it possible to obtain a direct correlation with the gravitational derived field.

CHAPTER II

TWO DIMENSIONAL CONVOLUTION AND SPECTRAL ANALYSIS

Introduction

The potential field data f(x,y) is considered as a mixture of random noise and aperiodic signals which constitute a stationary random process. The contribution by the near surface geological or topographical features to the potential field often fluctuates from place to place in a random manner. The effect due to such superimposed potentials may be considered as noise for our purposes. On the other hand an aperiodic signal implies a transient waveform of arbitrary shape which may be represented as a superposition of a continuous range of harmonics. The potential field measurements may thus consist of an ensemble of aperiodic signals produced by various sources in the spatial domain. By the 'stationary random process' it is implied that the mean and the covariance of the noise and aperiodic signals at two different places are space invariant or are unrelated.

The spectral decomposition of the potential field data f(x,y) can be represented by the following complex Fourier transform relations in the digital domain.

$$F(k_1,k_2) = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \; e^{\frac{2\pi i}{N}(k_1x+k_2y)} \qquad \ldots\ldots (2.1)$$

$$\text{for} \quad k_1,k_2 = 0,1,2,\ldots.,(N-1)$$

and conversely

$$f(x,y) = \frac{1}{N^2} \sum_{k_1=0}^{N-1} \sum_{k_2=0}^{N-1} F(k_1,k_2)\; e^{\frac{-2\pi i}{N}(k_1x+k_2y)} \quad \ldots\ldots (2.2)$$

for $x,y = 0,1,2,\ldots\ldots,(N-1)$

Where $F(k_1,k_2)$ are N complex-valued vectors. The wavelengths λ_1 and λ_2 are given by $\frac{2\pi}{k_1}$ and $\frac{2\pi}{k_2}$ where k_1 and k_2 are the wave numbers along x and y directions.

The function f(x,y) may be expanded in terms of a double trigonometric series in the interval (0,N-1 ; 0,N-1) and it has been shown in various texts (e.g. Lee 1960) that the expansion tends to the Fourier integral as N goes to infinity. The main difference between the Fourier series and the integral transform representation is that in the use of the series there is an assumption of an infinite series of identical patterns and hence a fundamental wave to which the harmonics are related whereas in the case of transforms the fundamental component has been extended to cover all space.

To analyse the potential field data, use is made of the fast Fourier transform algorithm which has been developed by Good (1958) and modified by Cooley and Tukey (1965), Gentleman and Sande (1966). Actually this method of computation was suggested first by Runge (1903, 1905) but was

overlooked by all except a few research workers using Fourier transforms for x-ray analysis. A brief discussion of the fast Fourier transform (F.F.T.) algorithm along with the necessary Fortran IV programs is given in Appendix A.

Power density spectrum

Using F.F.T. programs the potential field data may be represented in the wave number domain for power spectral studies (Appendix A). The power density function yields information on the amplitude of the dominant harmonics of which the data is composed. A study of the power spectra was carried out using a 200 mile long aeromagnetic profile from Northern Saskatchewan (figure 2.1) which was digitized at a ¼ mile interval. A very small interval was deliberately chosen to evaluate the effect of aliasing. The aeromagnetic data was obtained with a magnetometer flown 1000 feet above the surface on lines spaced ½ mile apart. The spectral plot is illustrated in figure (2.2 a). A pronounced inflection point is noticed at a wavelength of 5 miles with a power density $4 \times 10^4\ \gamma^2\ \text{mile}^{-1}$. It is concluded that the maximum power is concentrated at longer wavelengths, whereas at shorter wavelengths there are quite a few spectral highs which may be either due to very small scale geological features or noise due to digital errors. At the Nyquist wavelength $\lambda_{NYQ} = 0.5$ mile, the power density is of the order of $40\ \gamma^2\ \text{mile}^{-1}$ and one can conclude that

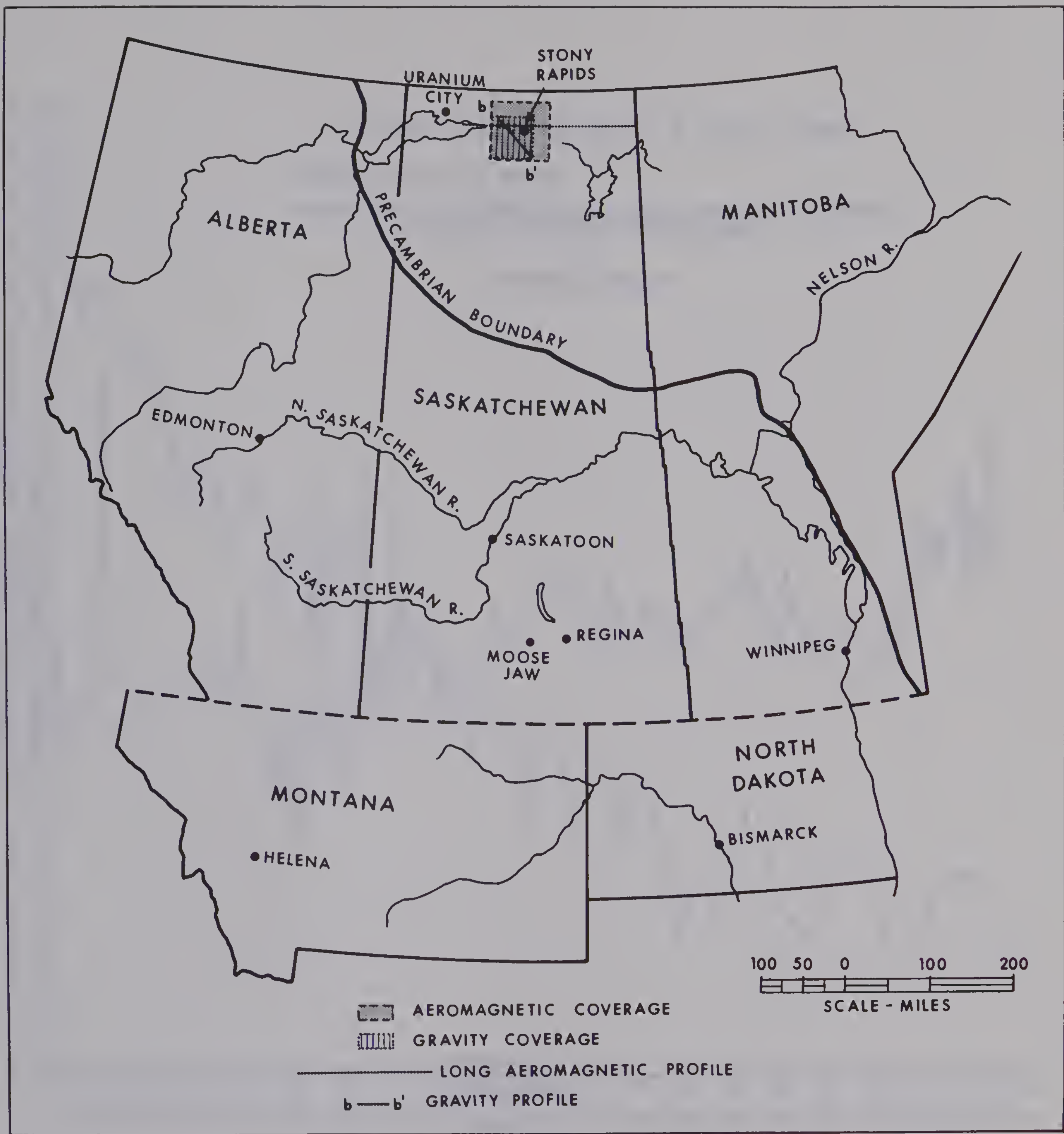


Figure 2.1

Location map of the study area

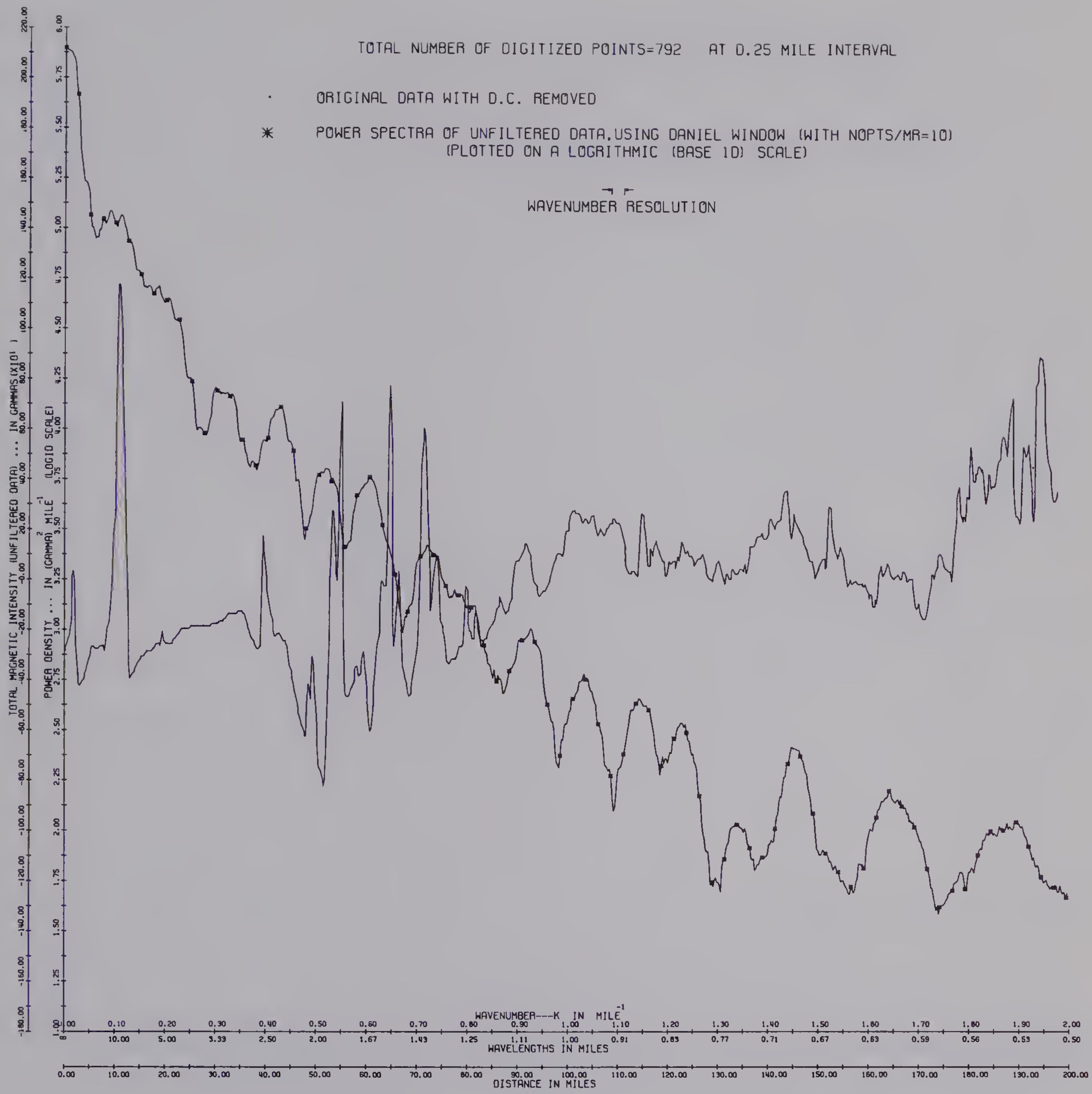


Figure 2.2 (a)

Spectral density plot of a long magnetic profile at ¼ mile digital interval

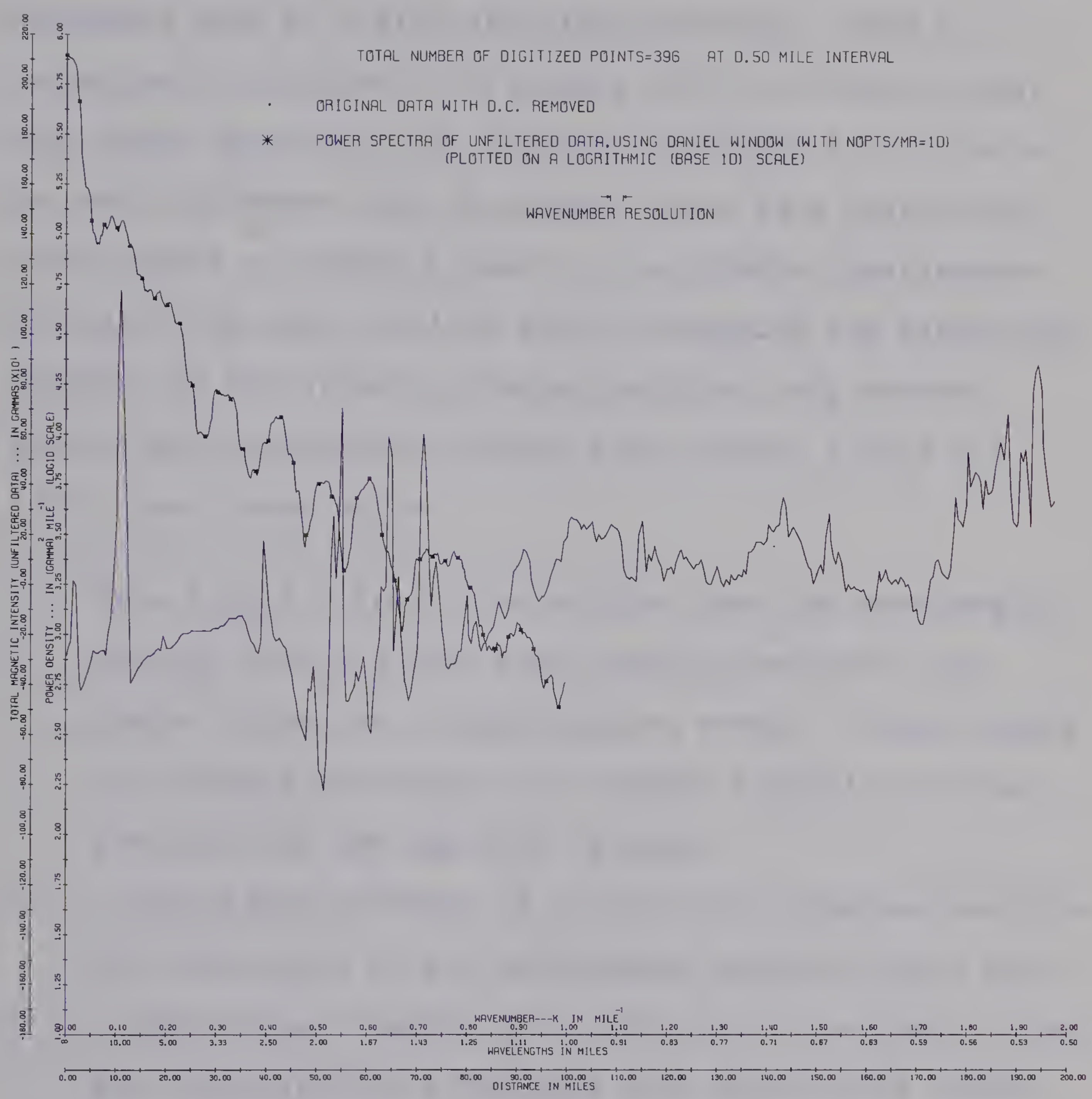


Figure 2.2 (b)

Spectral density plot of a long magnetic profile at ½ mile digital interval

any aliasing or folding back of the energy is negligibly small. Figure 2.2(b) illustrates the power spectra for the decimated data at ½ mile digitized interval. From a comparison of figures 2.2(a) and 2.2(b) it is quite clear that power density at the Nyquist wavelength of 1 mile is not very different, and in general there is a relatively small amount of energy present at the shorter wavelengths. One has to be very cautious about increasing the digitizing interval as the aliasing problem could be very severe. Some of the conclusions derived from figures 2.2(a) and 2.2(b) are listed below.

(1) From figure 2.2(a) it is noticed that for wavelengths shorter than 0.8 mile power density reflects the random noise due to digitization errors. These should be filtered out before any further analysis or interpretation of the map data is made.

(2) A digitizing interval of ½ mile will introduce negligible distortion at all wavelengths greater than 1 mile.

(3) A digitizing interval of 1 mile will be adequate if we are interested in structures with wavelengths longer than 4 miles. Since the energy is 10 times larger at a 4 mile wavelength than at the λ_{NYQ} = 2 mile.

(4) A digital interval as large as 2 miles is not recommended since the power density at the Nyquist wavelength of 4 miles is substantial and will certainly introduce

some distortion at longer wavelengths.

From this study it was concluded that a ½ mile digitizing interval was a reasonable compromise between the conflicting limitations imposed by aliasing, size of the core memory and the type of structures which might eventually be studied. A smaller digital interval may be desirable since it enables us to study small anomalies which may be of economic significance. On the other hand one is faced with computer limitations in analysing enormous amounts of data and a very difficult and tedious digitization problem.

<u>Digitization of two-dimensional data</u>

For computations it is necessary to digitize the data from the maps at a suitable sampling interval Δx. This was carried out manually by superimposing a grid on the field maps and reading the points at regular interval into a dictaphone. From this the data was transferred to IBM cards and subsequently to tape. To keep the aliasing errors to a minimum it is best to choose the digitizing interval, Δx, sufficiently small and subsequently filter the data to remove the wavelengths outside the range of interest. In the present study, the digitization of the aeromagnetic maps in the area (figure 2.1) with a sampling interval of ¼ mile would require about 80000 data points. From the above mentioned power density study, it seems quite adequate to

digitize the maps at ½ mile sampling intervals and reduce the number of data points to 20000. The labour of handling the data and the computation time is still substantial but since our intent, for the present, is to study the broad scale structures, this will be reduced after suitable smoothing of the data at short wavelengths. A portion of the study area is also covered with a detailed gravity survey conducted by the Gravity Division of the Dominion Observatory at an interval of 1 to 2 miles using helicopters. The writer participated in this field survey and made a collection of samples for density measurements. The gravity map was digitized at one mile intervals. Before digitizing the gravity map, the contours at the edges, where the data was sparse, were smoothed by hand.

Convolution and Decimation

The effect due to unwanted short or long wavelengths can be suppressed by applying the appropriate filters to the digitized data. In the present study box type of filters, with symmetrical and zero phase shift characteristics, were applied. The filter coefficients for low-cut, high-cut or bandpass operators were solved using equations that are well known but are briefly reviewed in Appendix B. To increase the reliability of the boundary data, it was considered feasible to 'taper' each end of the rows and columns of the matrix by means of a cosine bell

$$W(J) = \tfrac{1}{2}+\tfrac{1}{2}\cos\frac{\pi J}{L}$$

where $J = 1,2,.....,L$

and L = one half of the number of coefficients in the filter operator.

It was noticed that the boundary data did improve as compared to the results without tapering, but it is not critically important.

The impulse response and transfer function for a few of the two-dimensional filters used in this study are illustrated in figures 2.3(a), 2.3(b), 2.4(a) and 2.4(b). For example, figure 2.3(b) shows the NE quadrant of the transfer function due to a low-cut filter with cut-off at 2 miles and figure 2.4(b) represents the function due to a low-cut filter cut-off at 4 miles. Note that the response has a square pattern over the matrix and there is a strong attenuation along the NE-SW and NW-SE directions, as compared to the east or north directions. These properties arise from the technique used in implementing the convolution which consists of filtering along the rows and then a similar operation along the columns. The small amount of oscillations termed Gibb's phenomena is due to the truncation of the box car function at 25 terms. A filter with greater symmetry would be easy to design in the wavelength domain but for the present study these relatively simple filters were considered

adequate. The convolution of two-dimensional data with the filter operators and the decimation of the filtered data follow the steps in the block diagram shown in figure 2.5. Figures 2.6 and 2.7 show the original magnetic and gravity maps with D.C. removed and figures 2.8 and 2.9 illustrate the respective filtered maps with attenuation of wavelengths shorter than 4 miles. Other filtered maps are attached in Appendix B. In this study the digital interval was reduced to 1 mile to have a grid of 69 by 69 matrix for the magnetic map or to a 2 mile interval in which case the digitized data is reduced to a 35 by 35 matrix.

Upward and Downward Continuation

Smoothing or convolution is also produced by means of upward and downward continuation, first, second and higher derivatives which suppress some of the characteristics of the data and reveal others which are not too apparent in the original data. In accordance with equation (1.1) the continuation of the potential field data is easily performed by using the fast Fourier transforms. Once the complex Fourier coefficients are obtained, it is possible to perform upward or downward continuation of the field, at any height or depth by substituting the desired value of Z and taking the inverse fast Fourier transform to return to the spatial domain. It is noted that the continuation process is valid only in the space without sources, if this condition is

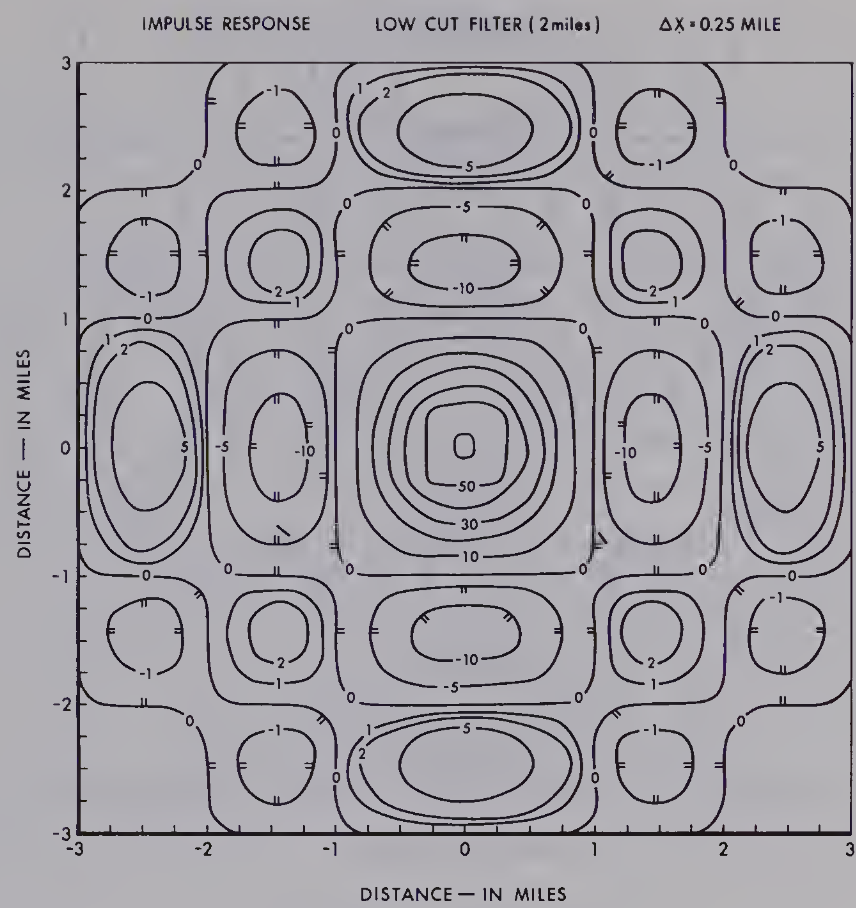


Figure 2.3 (a)

Impulse response of a two-dimensional low-cut filter (which cuts-off wavelengths shorter than 2 miles)

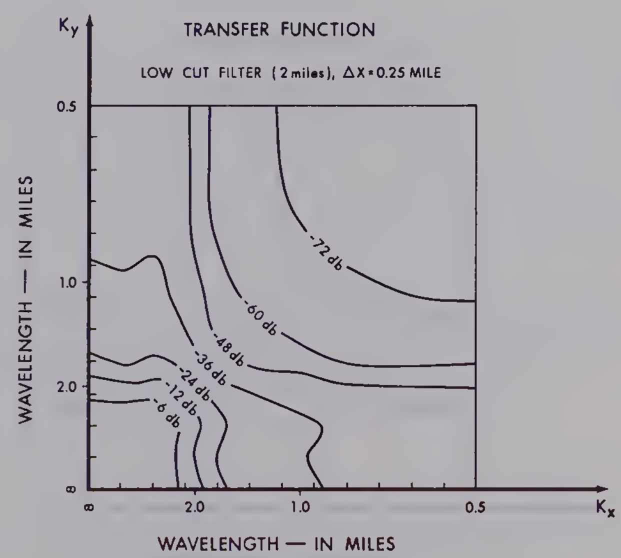


Figure 2.3 (b)

NE quadrant of a transfer function for the case of figure 2.3 (a)

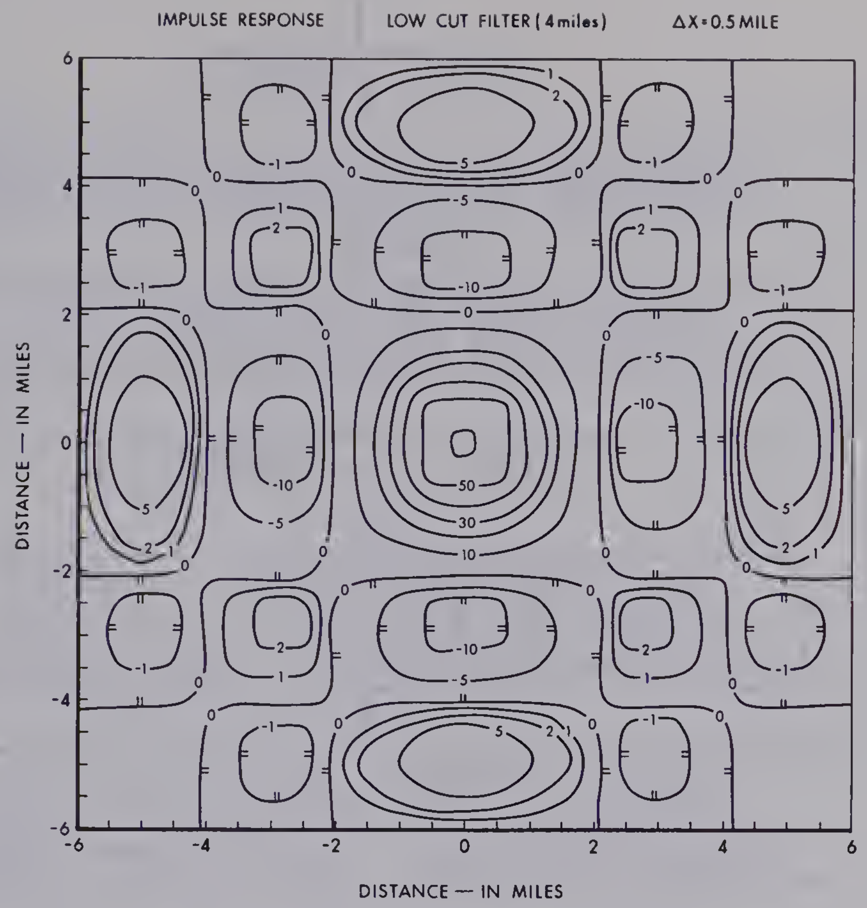


Figure 2.4 (a)

Impulse response of a two-dimensional low-cut filter (which cuts-off wavelengths shorter than 4 miles)

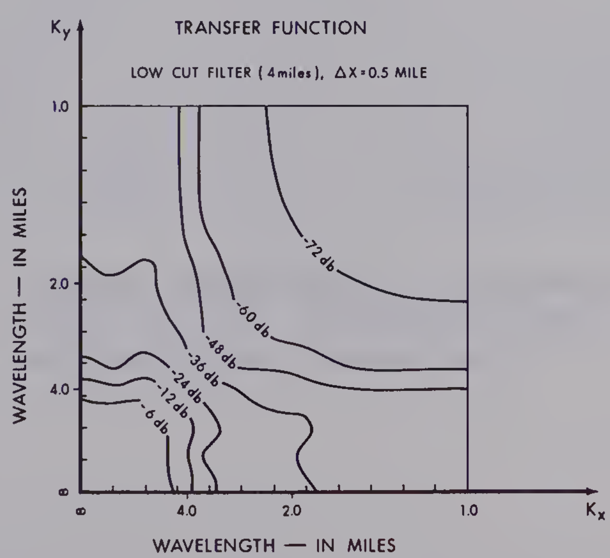


Figure 2.4 (b)

NE quadrant of a transfer function for the case of figure 2.4 (a)

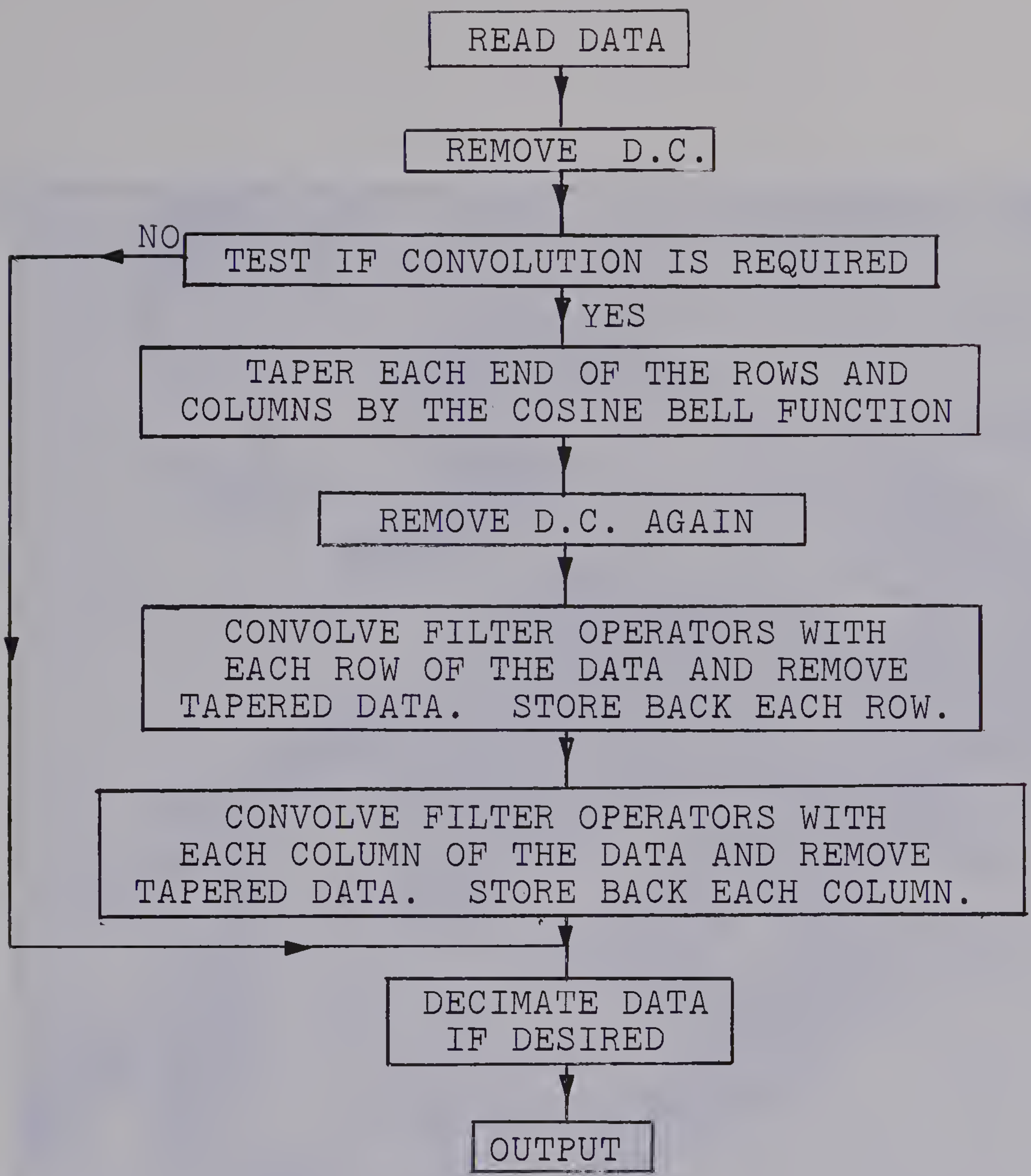


Figure 2.5. Block diagram for convolution of the filter operators with the two-dimensional data set.

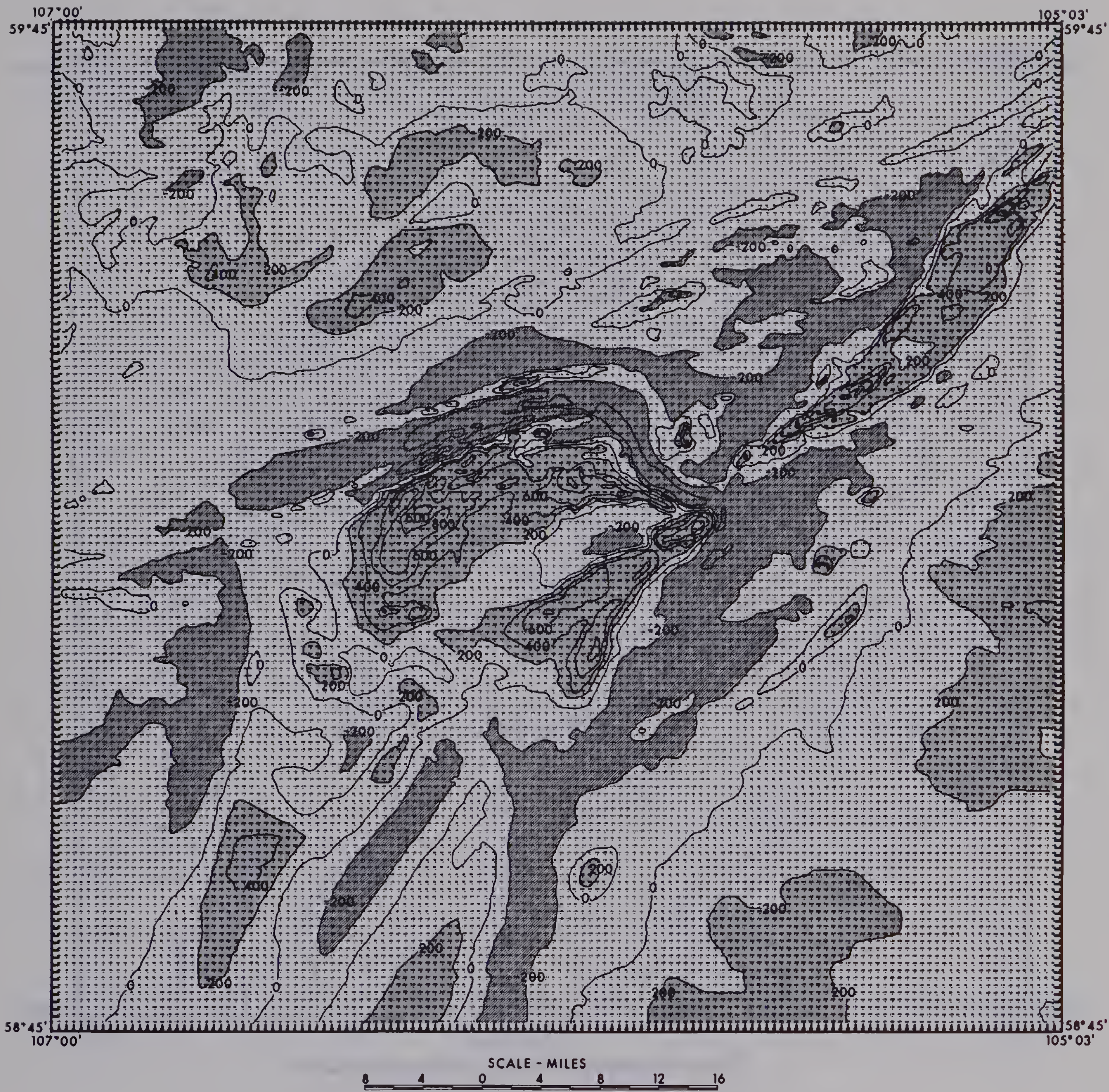


Figure 2.6

Original magnetic map with D.C. removed

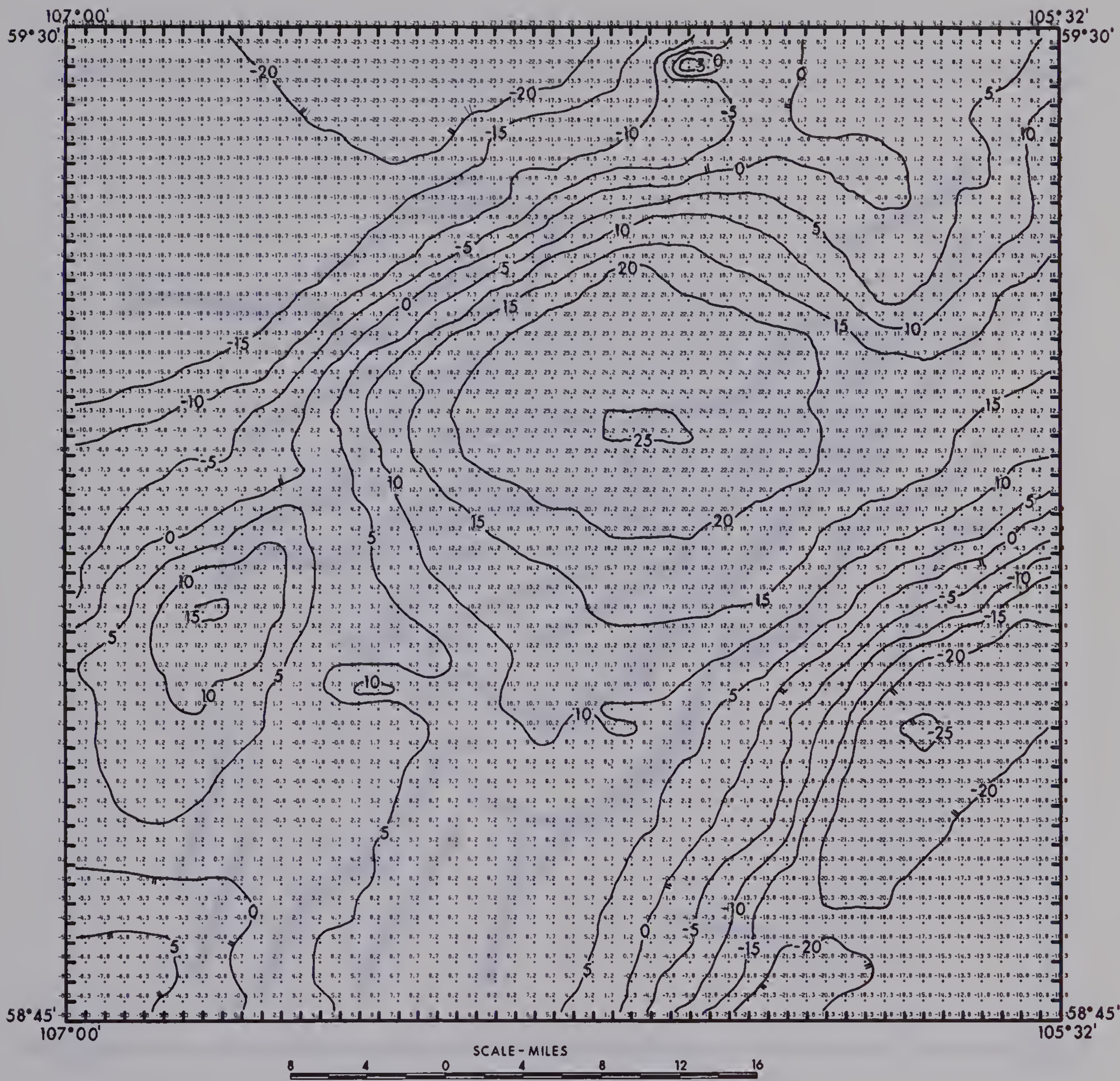


Figure 2.7

Original gravity map with D.C. removed

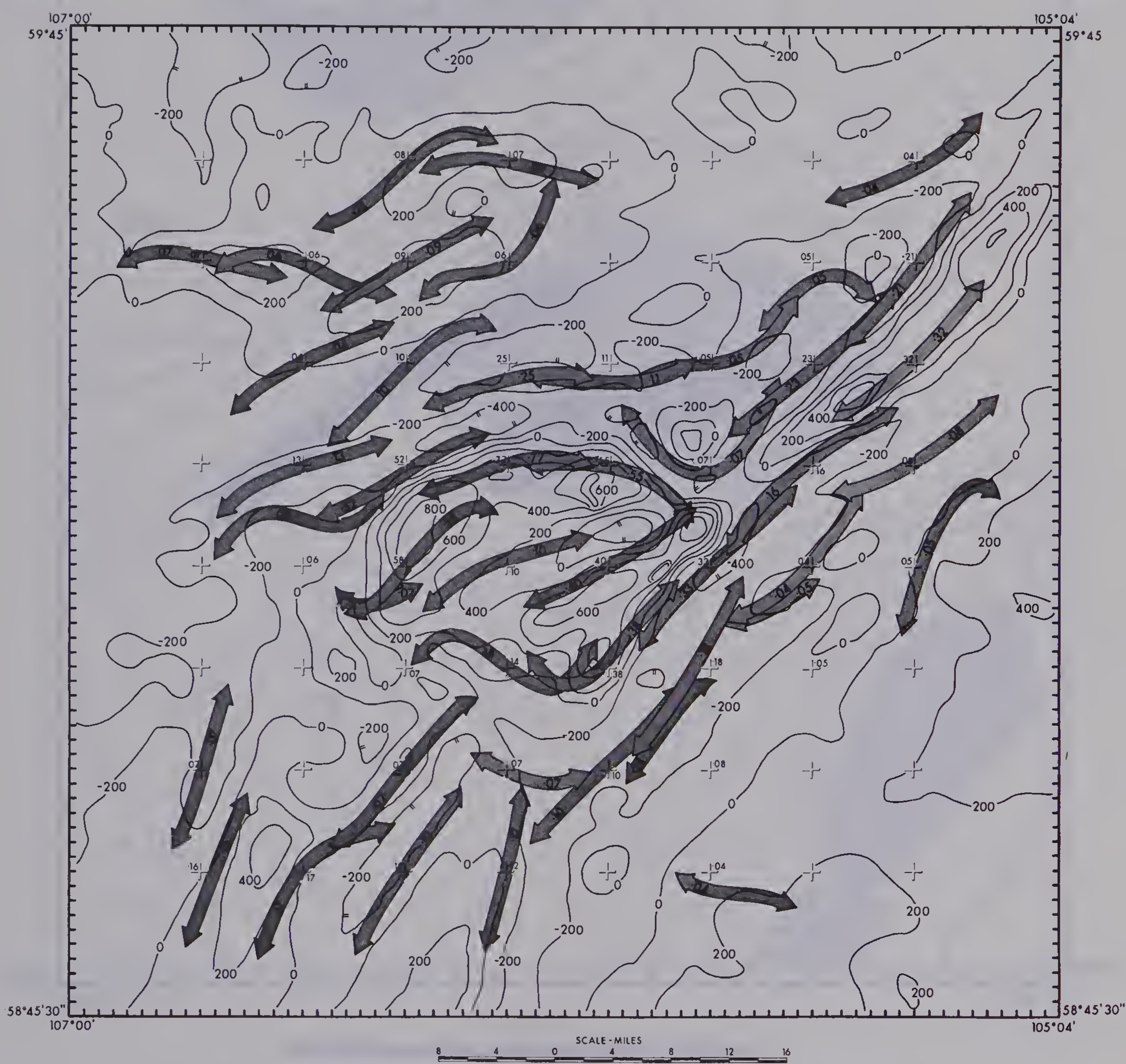


Figure 2.8

Filtered magnetic map with attenuation of wavelengths shorter than 4 miles. (trends are also shown as obtained by using a test map of 6 by 6 miles, see Chapter III.)

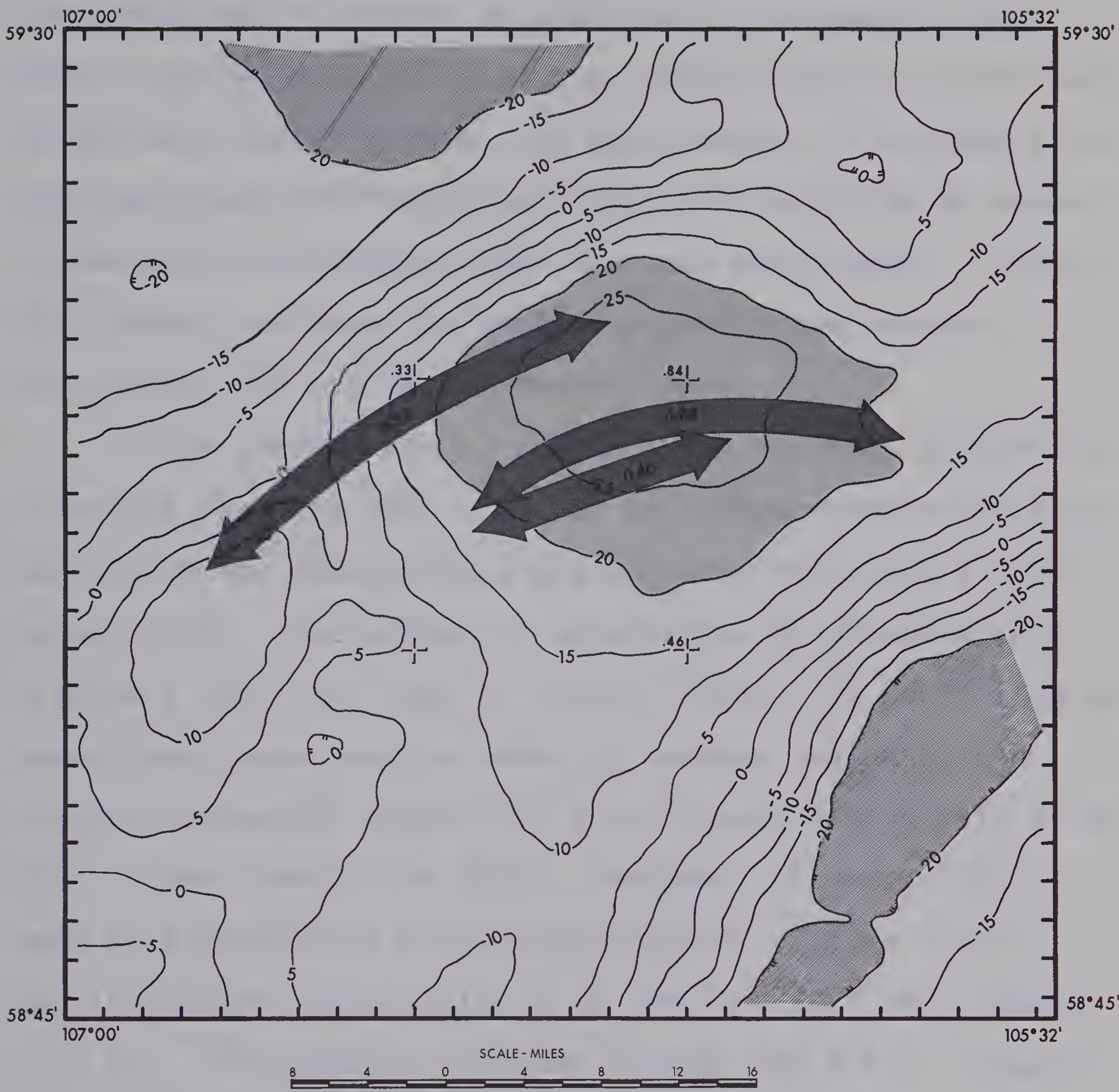
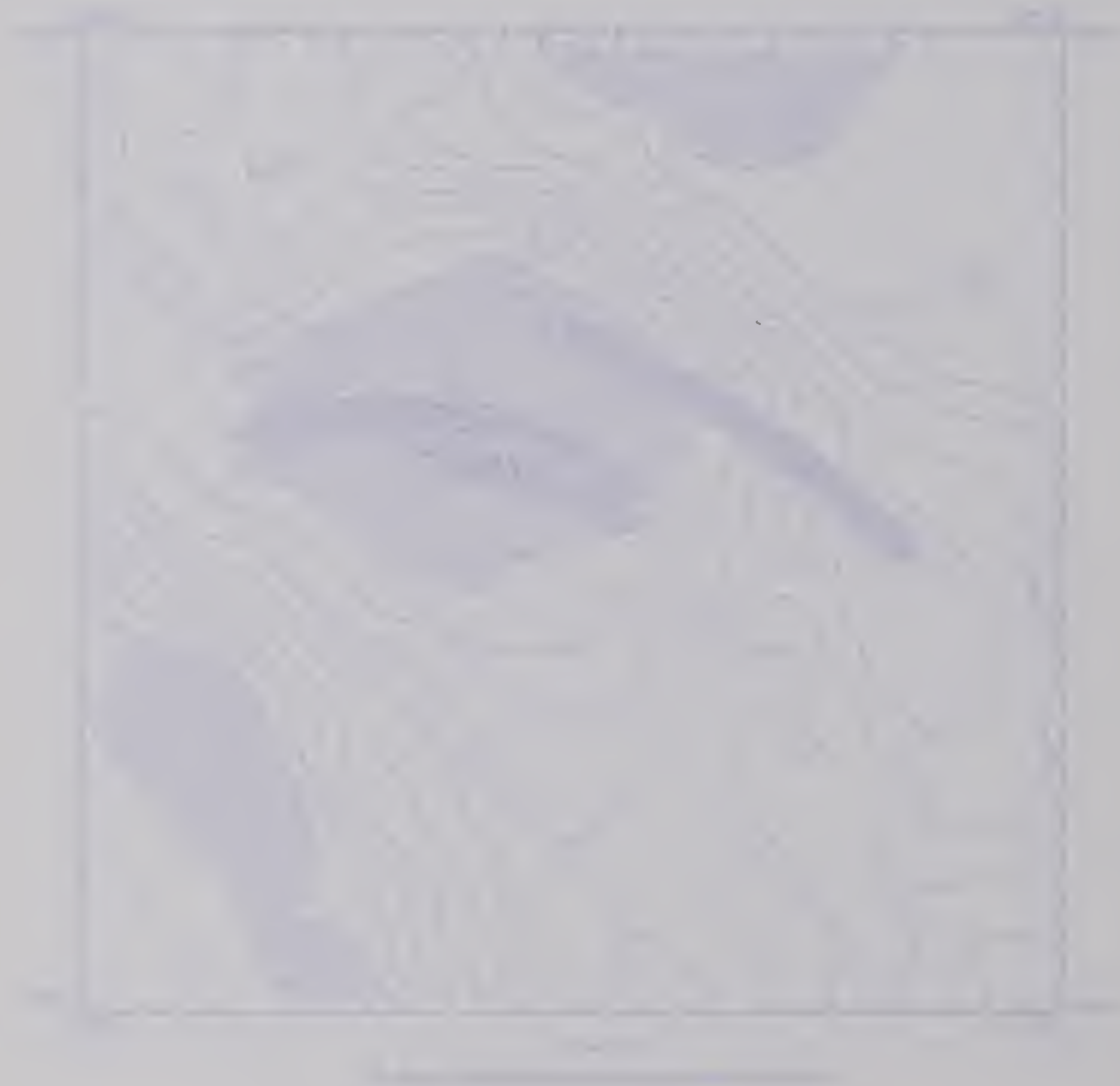


Figure 2.9

Filtered gravity map with attenuation of wavelengths shorter than 4 miles. (trends are also shown as obtained by using a test map of 12 by 12 miles, see Chapter III)



violated the field map will show violent oscillations and a general instability. From equation (1.1) it is seen that the continuation process is effected by the multiplying factor $\exp\{\frac{2\pi Z}{L}(m^2+n^2)^{\frac{1}{2}}\}$ with the complex Fourier coefficients in the wave number domain. In this factor, it is also clear that the short wavelengths are much more affected by upward or downward continuation than the long wavelengths. Therefore upward continuation tend to suppress the shorter wavelengths and act as a lowpass filter.

In the present study, continuation has been achieved by arranging the data into a matrix with even symmetry as this simplifies the calculations and provides the results in a compact form. The method of calculation is substantially different from that used by Tsuboi (1959). In performing the operations, care must be taken to arrange the data in a properly symmetric fashion in a two dimensional matrix to be able to make use of the F.F.T. programs. Tsuboi (1959) gave an example of a surface gravity map (figure 2.10 a) and its upward continuation at a level of 15.9 km. (figure 2.10 b). The results obtained through the F.F.T. programs are in excellent agreement (figure 2.10 c).

Derivatives

It is also possible to obtain the N^{th} order vertical derivative simply by multiplying the factor $\{\frac{2\pi}{L}(m^2+n^2)^{\frac{1}{2}}\}^N$

with the complex Fourier coefficients in the wave number domain and then taking the inverse Fourier transforms of the subsequent coefficients. Figure 2.10(d) shows the first vertical derivative map as obtained using the F.F.T. programs. Further details and various Fortran IV programs pertaining to these techniques are listed in Appendix C.

Summary

From a power spectral study of a long magnetic profile the sampling interval was established for digitizing aeromagnetic maps in the Shield area of Saskatchewan. The digitized magnetic data covers an area of 5000 square miles and consists of 20000 points on a grid of ½ mile spacing. A part of this area is also digitized at a ¼ mile interval to study the effects introduced by this fine grid. The effect of aliasing errors on structures of various wavelengths was considered. Since the intent in the study is broad scale regional structures, the data was convolved with coefficients of a low-cut operator to remove wavelengths shorter than 4 miles. The region in which there is a common gravity and magnetic coverage is an area of about 2500 square miles with a digitizing interval of 2 miles after decimation, the data consisting of a matrix of 26 x 26 points. Various filtered maps were illustrated and it was shown that the fast Fourier program may be used to obtain upward and downward continuation and derivative maps in an efficient manner.

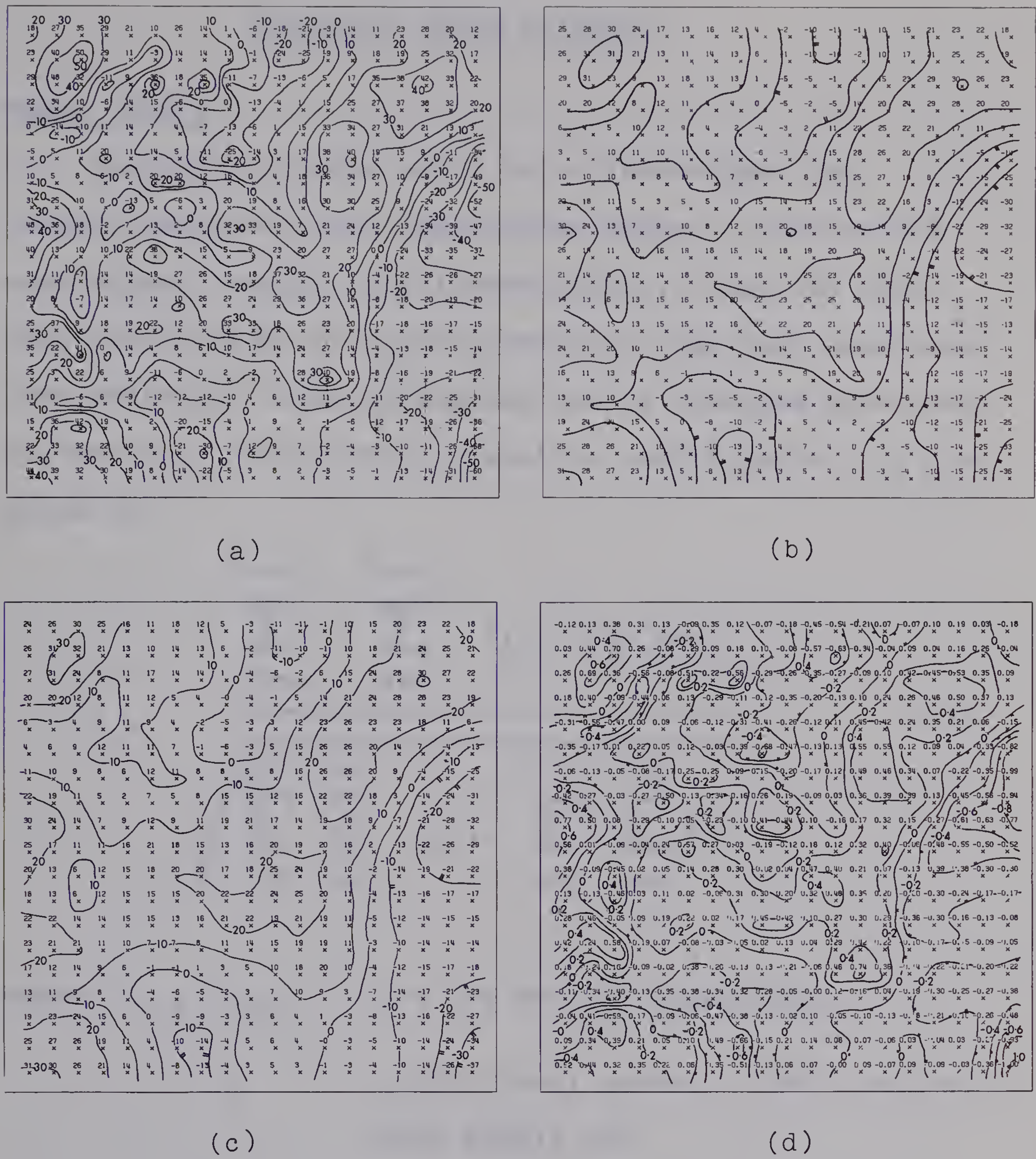


Figure 2.10

(a) Surface gravity map (after Tsuboi 1959)

(b) Upward continuation of (a) at a level of 15.9 Km. above the surface (after Tsuboi 1959)

(c) Upward continuation of (a) at a level of 15.9 Km. above the surface obtained by using fast FOURIER transform programs

(d) First vertical derivative map of (c) obtained by using F.F.T. programs

CHAPTER III

AUTOMATIC TREND ANALYSIS

Basic Theory

The object of the study is to investigate the possibilities of trend analysis by means of statistical techniques. During this investigation, a new and somewhat novel method of cross-correlation has been developed from which the trend directions can be obtained with some reliability. The cross-correlation coefficients $C_{r,s}$ are given by:

$$C_{r,s} = \frac{\sum_{j=0}^{N-r-1} \sum_{\ell=0}^{N-s-1} X_{j,\ell}\, Y_{j+r,\ell+s}}{\sqrt{\sum_{j=0}^{N-1} \sum_{\ell=0}^{N-1} X^2_{j,\ell} \sum_{j=0}^{N-1} \sum_{\ell=0}^{N-1} Y^2_{j,\ell}}} \qquad \ldots\ldots(3.1)$$

Where $X_{j,\ell}$, $Y_{j,\ell}$ are two sets of sample data.

N is the total number of data points in each sample set.

$r,s = 0,\pm1,\pm2,\ldots\ldots,\pm m$

m = maximum lag or displacement.

Normally, the cross-correlation is obtained on two different sets of sample data. However in this section

the same matrix of values will be used for both functions but one of them will be modified by surrounding the central core with a null field. Let $X_{j,\ell}$ be the digitized values for any two dimensional function and $Y_{j,\ell}$ is chosen in such a way that

$$\left.\begin{array}{llll} Y_{j,\ell} & = & X_{j,\ell} & \text{for} \quad |j|,\ |\ell| \leq n' \\ & = & 0 & \text{otherwise} \end{array}\right\} (n' < n) \qquad \ldots\ldots(3.2)$$

Where the positive integers n' and n are defined as shown in figure (3.1) with origin (0,0) at the centre of the matrix.

The sample map is defined as the N^2 complete set of values $X_{j,\ell}$ while the test map is defined as the N_H^2 non-zero values of the central core of $Y_{j,\ell}$. This convention of sample map and test map (figure 3.2) will be used throughout this chapter. The values on the test map are identical to the central core of data placed in $X_{j,\ell}$. However, the important concept of introducing zero's all around $Y_{j,\ell}$ makes it possible to obtain the correlation coefficients for the same sample set without the limitations imposed by the symmetry properties of an auto-correlation function. For the purposes of trend analysis the cross-correlation will be defined in the following manner:

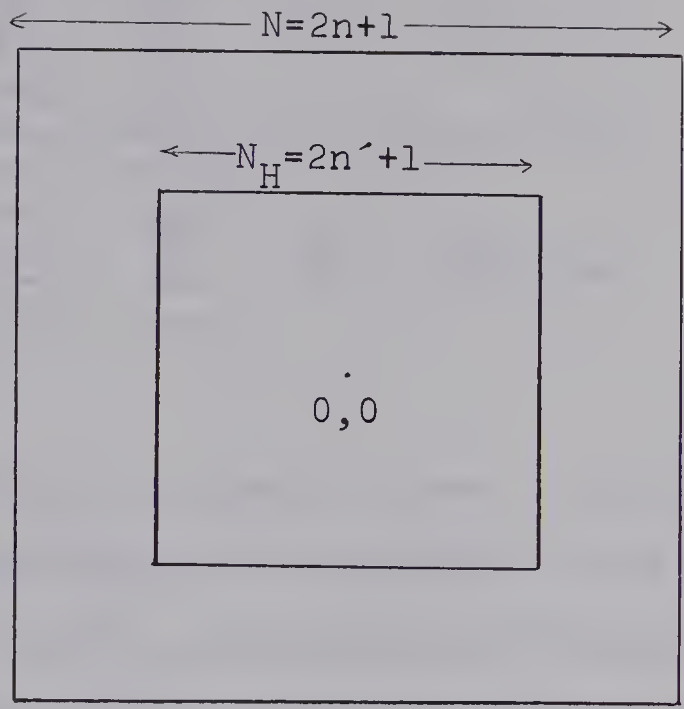


Figure 3.1. Conventions for choosing $y_{j,\ell}$ from $X_{j,\ell}$ as defined in equation (3.2)

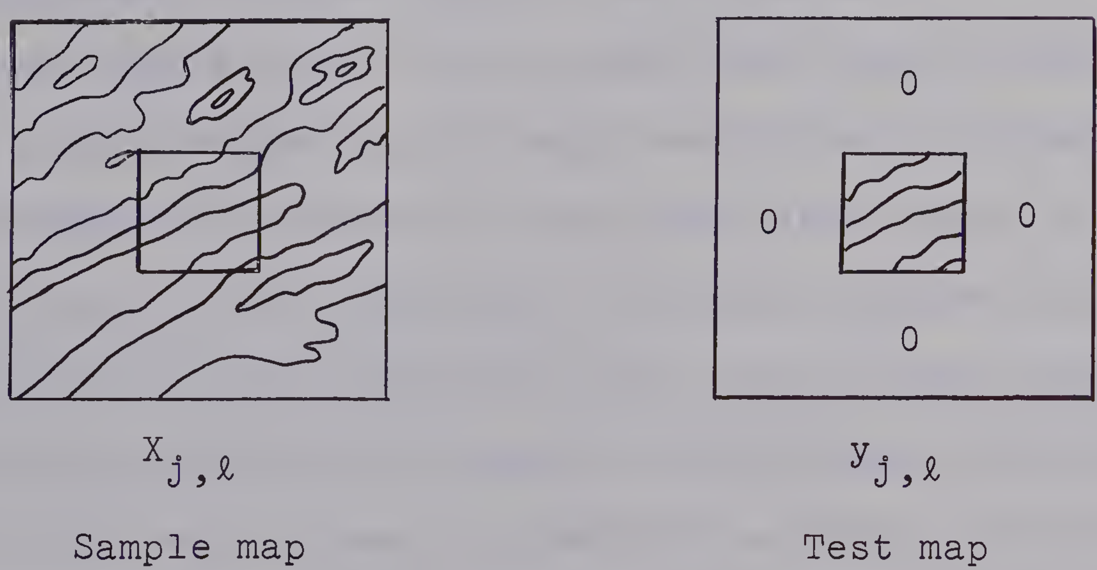


Figure 3.2 Schematic diagram for the sample sets $X_{j,\ell}$ and $y_{j,\ell}$ used for the new correlation technique.

$$C^{\mu}_{r,s} = \frac{\sum\limits_{j=0}^{N-r-1} \sum\limits_{\ell=0}^{N-s-1} X^{\mu}_{j,\ell} \; Y^{\mu}_{j+r,\ell+s}}{\sum\limits_{j=0}^{N-r-1} \sum\limits_{\ell=0}^{N-s-1} X^{\eta}_{j,\ell} \; Y^{\eta}_{j+r,\ell+s}} \qquad \text{......(3.3)}$$

The sample map $X^{\mu}_{j,\ell}$ is a small subset μ of a larger map from which trends are to be determined. The normalization is with respect to $C^{\eta}_{r,s}$, the particular subset of $\mu=\eta$, which yields the largest correlation coefficient as defined by equation (3.4). This normalization procedure may appear somewhat arbitrary but it insures that no correlation is greater than 1 over the entire map and the strongest trends appear with the highest coefficients.

From figure (3.6 c) it is seen that when the data sets have a high correlation, a large coefficient is obtained. The decrease or increase in the coefficient value is also indicative of trend direction. A similar pattern which propagates along any direction will yield a high positive correlation coefficient; negative correlations are possible and indicate where areas of positive curvature correlate with negative ones. From these coefficients, it is possible to track the primary or secondary trends. In the present study automatic digital determination was restricted to the primary trends.

Size of sample data sets for cross-correlation

If the large scale features are of interest the two-dimensional data is convolved with the filter operators to remove features of short wavelengths. After filtering, the data is decimated in order to reduce the number of coefficients to be calculated. The test map is an N_H by N_H size matrix in the centre of $Y_{j,\ell}$ and it together with the desired number of lags determines the size of the sample map. As an example, a 6 by 6 mile test map at a one mile digital interval may be cross-correlated with an 18 by 18 mile sample map for lags $0,\pm1,\pm2,\ldots,\pm6$ which will produce a 12 by 12 mile cross-correlation map. To obtain a sampling of cross-correlations at representative sites over a whole map, a new test map is chosen at equal displacement and the cross-correlation calculations are repeated over the entire field. The flow diagram in Appendix D outlines the procedure to obtain the sample and test map matrices.

Two-dimensional cross-correlation

Consider the covariance from equation (3.1) which can be written as

$$\mathrm{Cov}_{r,s} = \sum_{j=0}^{N-r-1} \sum_{\ell=0}^{N-s-1} X_{j,\ell}\, Y_{j+r,\ell+s} \qquad \ldots\ldots(3.4)$$

The direct evaluation of the coefficients using the above equation requires a large amount of computer time. By Wiener's theorem one can evaluate equation (3.4) by using the Fourier transform technique but this is also very slow unless a fast Fourier transform algorithm is used. The F.F.T. algorithm reduces the computing time and provides better accuracy than the standard Fourier transforms. This requires special care in preparing the input to the F.F.T. algorithm. Previously the one-dimensional case has been considered by Gentleman and Sande (1966) and the method has been extended here to the two dimensional case. It is thought appropriate to present this approach in the following section.

It is well-known that the Fourier transforms are cyclic in character and the correlation operations are non-cyclic. The difference may be explained by considering the two-dimensional matrix of 3 by 3 (figure 3.3 a). According to the cyclic property of the Fourier transforms this matrix is the same in all space and repeats itself as shown in (figure 3.3 b). In the correlation operations, a field of zero's is assumed to exist all around the data. This discrepancy between the cyclic character of the Fourier transforms and the non-cyclic operations of the correlation techniques can be removed by adding zero's to the actual physical data in such a manner that both the conditions are satisfied (figure 3.3 c).

$$\begin{array}{|ccc|}\hline a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \\ \hline \end{array}$$

(a)

$$\begin{array}{|ccc|ccc|ccc|}\hline a_{11} & a_{12} & a_{13} & a_{11} & a_{12} & a_{13} & a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} & a_{21} & a_{22} & a_{23} & a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} & a_{31} & a_{32} & a_{33} & a_{31} & a_{32} & a_{33} \\ \hline a_{11} & a_{12} & a_{13} & a_{11} & a_{12} & a_{13} & a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} & a_{21} & a_{22} & a_{23} & a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} & a_{31} & a_{32} & a_{33} & a_{31} & a_{32} & a_{33} \\ \hline a_{11} & a_{12} & a_{13} & a_{11} & a_{12} & a_{13} & a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} & a_{21} & a_{22} & a_{23} & a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} & a_{31} & a_{32} & a_{33} & a_{31} & a_{32} & a_{33} \\ \hline \end{array}$$

(b)

$$\begin{array}{|ccc|ccc|ccc|}\hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & a_{11} & a_{12} & a_{13} & 0 & 0 & 0 \\ 0 & 0 & 0 & a_{21} & a_{22} & a_{23} & 0 & 0 & 0 \\ 0 & 0 & 0 & a_{31} & a_{32} & a_{33} & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline \end{array}$$

(c)

Figure 3.3 (a) Original matrix data set.

(b) Presentation of the data in (a) for a cyclic operation such as in FOURIER transforms.

(c) Superposition of a box car function around the data in (a) for a non-cyclic operation.

Mathematically, if we extend the summation in equation (3.4) to $(N'-1)$ with the assumption that

$$\left.\begin{aligned} X_{j,\ell} = 0,\ Y_{j,\ell} = 0 \quad \text{for either} \quad & j = N, N+1, \ldots\ldots, N'-1 \\ \text{or} \quad & \ell = N, N+1, \ldots\ldots, N'-1 \end{aligned}\right\} \qquad \ldots\ldots(3.5)$$

and

$$\left.\begin{aligned} N'-1 &= N-1+|r|_{max} = 2N-1 \\ N'-1 &= N-1+|s|_{max} = 2N-1 \end{aligned}\right\} \qquad \ldots\ldots(3.6)$$

then from equation (3.4) we have

$$Cov_{r,s} = \sum_{j=0}^{N-r-1} \sum_{\ell=0}^{N-s-1} X_{j,\ell} \; Y_{j+r,\ell+s}$$

$$= \sum_{j=0}^{N-r-1} \sum_{\ell=0}^{N'-1} \sum_{p=0}^{N'-1} X_{j,\ell} \; Y_{j+r,p} \; \delta_{N'}[p-(\ell+s)]$$

or

$$Cov_{r,s} = \sum_{j=0}^{N'-1} \sum_{\ell=0}^{N'-1} \sum_{p=0}^{N'-1} \sum_{q=0}^{N'-1} X_{j,\ell} \; Y_{p,q} \; \delta_{N'}[p-(\ell+s)] \times \delta_{N'}[q-(j+r)] \qquad \ldots\ldots(3.7)$$

Where $\delta_{N'}[p-(\ell+s)]$ and $\delta_{N'}[q-(j+s)]$ are Kronecker delta functions with their argument being considered as modulo N' that is

$$\delta_{N'}(KN') = 1 \quad \text{for K being integer} \qquad \ldots\ldots(3.8)$$
$$= 0 \quad \text{otherwise}$$

Using the orthogonality property of the Fourier transforms (Gentleman and Sande 1966) which states that

$$\sum_{t=0}^{N'-1} e^{\frac{2\pi i t(\hat{t}-\hat{t}')}{N'}} = N'\delta_{N'}(\hat{t}-\hat{t}') \qquad \ldots\ldots(3.9)$$

equation (3.7) can be written as:

$$\mathrm{Cov}_{r,s} = \frac{1}{N'N'} \sum_{j,\ell} \sum_{p,q} X_{j,\ell}\, Y_{p,q} \times$$
$$\sum_{k_1=0}^{N'-1} \sum_{k_2=0}^{N'-1} e^{\frac{2\pi i k_1(q-j-r)}{N'}} e^{\frac{2\pi i k_2(p-\ell-s)}{N'}}$$

$$= \frac{1}{N'^2} \sum_{k_1,k_2=0}^{N'-1} \left[\sum_{j,\ell} X_{j,\ell}\, e^{\frac{-2\pi i k_1 j}{N'}} e^{\frac{-2\pi i k_2 \ell}{N'}} \right] \times$$
$$\left[\sum_{p,q} Y_{p,q}\, e^{\frac{+2\pi i k_1 q}{N'}} e^{\frac{+2\pi i k_2 p}{N'}} \right] e^{\frac{-2\pi i k_1 r}{N'}} e^{\frac{-2\pi i k_2 s}{N'}}$$

or

$$\mathrm{Cov}_{r,s} = \frac{1}{N'^2} \sum_{k_1=0}^{N'-1} \sum_{k_2=0}^{N'-1} \hat{X}_{k_1,k_2} \hat{Y}^*_{k_1,k_2} \, e^{\frac{-2\pi i k_1 r}{N'}} \, e^{\frac{-2\pi i k_2 s}{N'}}$$

......(3.10)

Where

$$\hat{X}_{k_1,k_2} = \sum_{j=0}^{N'-1} \sum_{\ell=0}^{N'-1} X_{j,\ell} \, e^{\frac{-2\pi i k_1 j}{N'}} \, e^{\frac{-2\pi i k_2 \ell}{N'}}$$

$$= \hat{a}_{k_1,k_2} + i \, \hat{b}_{k_1,k_2}$$

......(3.11)

and

$$\hat{Y}_{k_1,k_2} = \sum_{j=0}^{N'-1} \sum_{\ell=0}^{N'-1} Y_{j,\ell} \, e^{\frac{-2\pi i k_1 j}{N'}} \, e^{\frac{-2\pi i k_2 \ell}{N'}}$$

$$= \hat{c}_{k_1,k_2} + i \, \hat{d}_{k_1,k_2}$$

......(3.12)

Cross-power $\hat{P}_{k_1,k_2}$ is defined as:

$$\hat{P}_{k_1,k_2} = \hat{X}_{k_1,k_2} \hat{Y}^*_{k_1,k_2} = (\hat{a}_{k_1,k_2} + i \, \hat{b}_{k_1,k_2}) \, (\hat{c}_{k_1,k_2} - i \, \hat{d}_{k_1,k_2})$$

......(3.13)

Using equation (3.13) we can write equation (3.10) as:

$$\mathrm{Cov}_{r,s} = \frac{1}{N'^2} \sum_{k_1=0}^{N'-1} \sum_{k_2=0}^{N'-1} \hat{P}_{k_1,k_2} \, e^{\frac{-2\pi i k_1 r}{N'}} \, e^{\frac{-2\pi i k_2 s}{N'}} \qquad \ldots\ldots(3.14)$$

Where $r,s = 0,\pm1,\pm2,\ldots,\pm m$ and $N' \geq N+m$

Usually, in using the F.F.T. we have taken the maximum lag m equal to N. Thus it is possible to obtain the cross-correlation coefficients by using equation (3.14) with the F.F.T. algorithm, in a simple manner.

From equation (3.5) it is to be noted that the actual data sets of size N by N must be stored under the dimensions N' by N' where $N' \geq 2N$ for N desired lags. In other words, the data is read in from row 1 to N and columns from 1 to N and the rows from (N+1) to N' and columns from (N+1) to N' are filled with a field of zero's as shown in figure 3.4.

The procedure to obtain the normalized coefficients is shown in the block diagram (figure 3.5), and the Fortran IV program to obtain the coefficients is listed in Appendix D.

$$\boxed{\begin{matrix} a_{11} & a_{12} & a_{13} & 0 & 0 & 0 \\ a_{21} & a_{22} & a_{23} & 0 & 0 & 0 \\ a_{31} & a_{32} & a_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{matrix}}$$

Figure 3.4. Setting up the original data in figure 3.3 (a) for calculating the cross-correlation coefficients using F.F.T. programs.

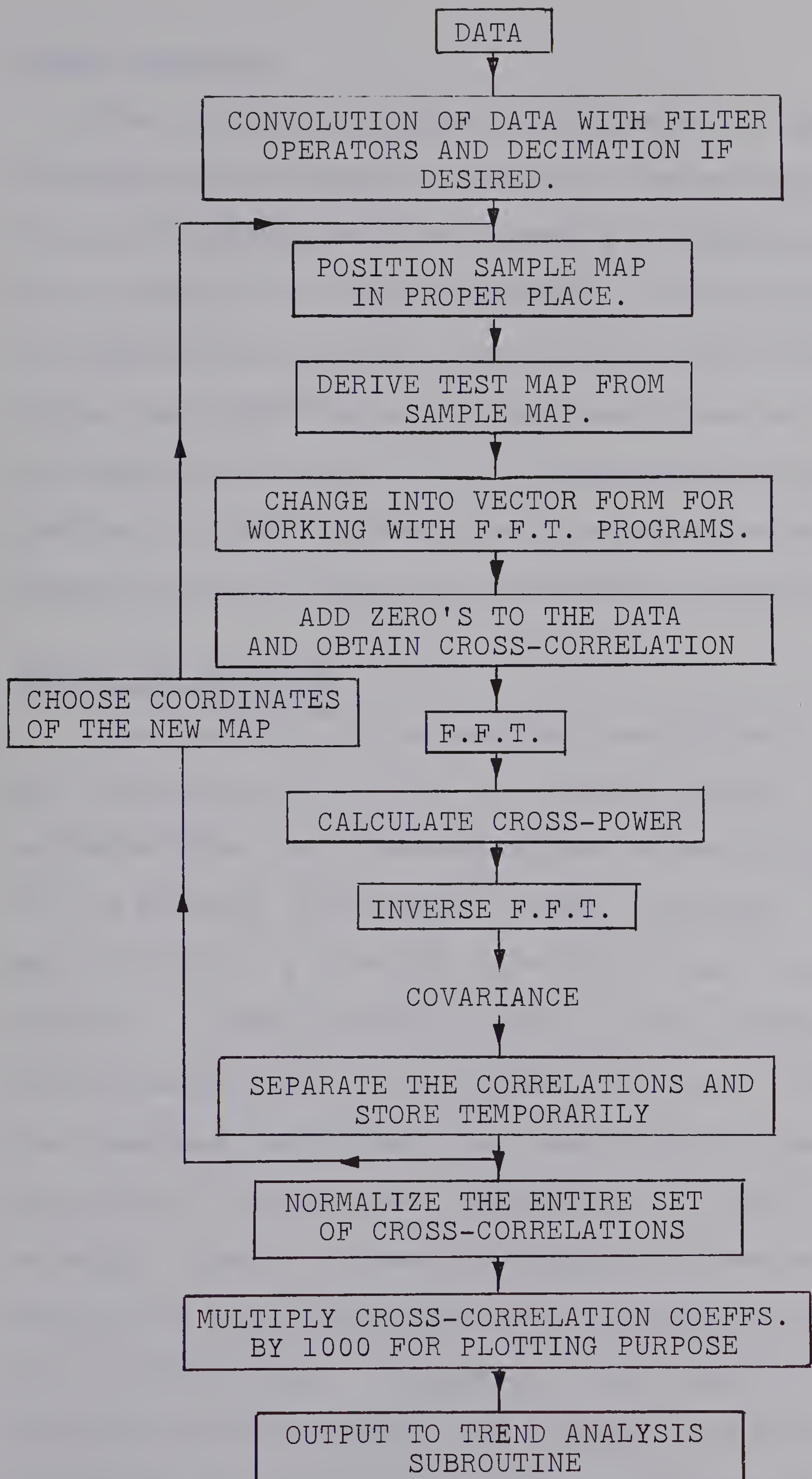


Figure 3.5. Block diagram for cross-correlation coefficients.

Trend Analysis

The cross-correlation coefficients, in general, vary systematically between +1 and -1. The spatial variation of the coefficients provides a method of tracing the trends due to various geological factors. In the study area, where the magnetic or gravity features have high cross-correlation, the coefficients obtained are close to +1 otherwise the coefficients are low. An empirical method has been devised so that the trends can be traced in an unbiased manner by using the cross-correlation coefficients.

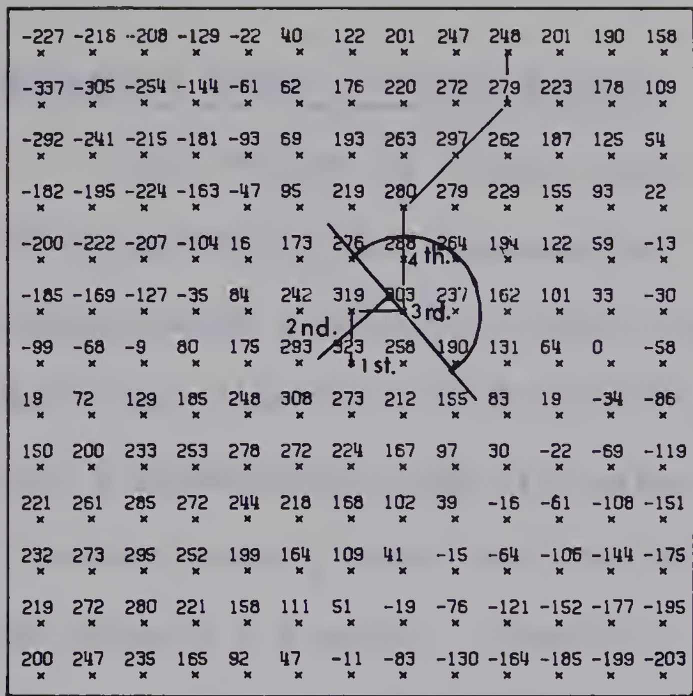
Method of Analysis

The location of the maximum coefficient is chosen for the starting point. Then the second maximum coefficient is selected from the 8 nearest points surrounding the location of the maximum coefficient. After obtaining the two adjacent maxima points, a straight line fit is made through these points. A normal drawn to the line and passing through the second maxima point is obtained and a scan is made of the next maximum coefficient in a semi-circle along the forward direction. To find the fourth point, a least squares straight line fit is made through the three points selected previously with weighting factors proportional to the correlation coefficients. A normal to this line is drawn to pass through the third maxima and a search for the fourth one is continued in a semi-circle along the forward direction. The

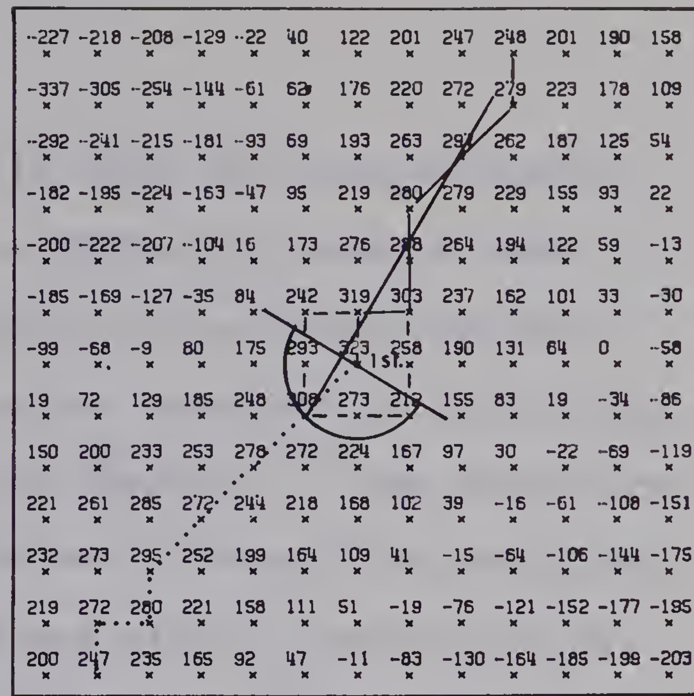
expressions for the least squares straight line fit and the normal to be drawn are given in Appendix E.

The program avoids arbitrarily sharp changes in the trend direction by using a maximum of seven previously determined trend positions, if these are available, to determine the direction of the straight line for the eighth and subsequent trend points. The procedure of determining a new straight line fit and drawing the normal to scan in a semicircle in a particular direction continues until the boundary is reached. At this point, the control is transferred to the first maximum point for scanning in the opposite direction. The process to be followed is indicated in figure (3.6 a and b). At the end of scanning a least squares third degree polynomial equation is fitted to the selected sites, with weights given by the correlation coefficients, to obtain a smooth trend line (figure 3.6 c). The expression for the third degree polynomial least squares fit is given in Appendix E. The scheme to be followed in tracing the trends is shown in the block diagram (figure 3.7).

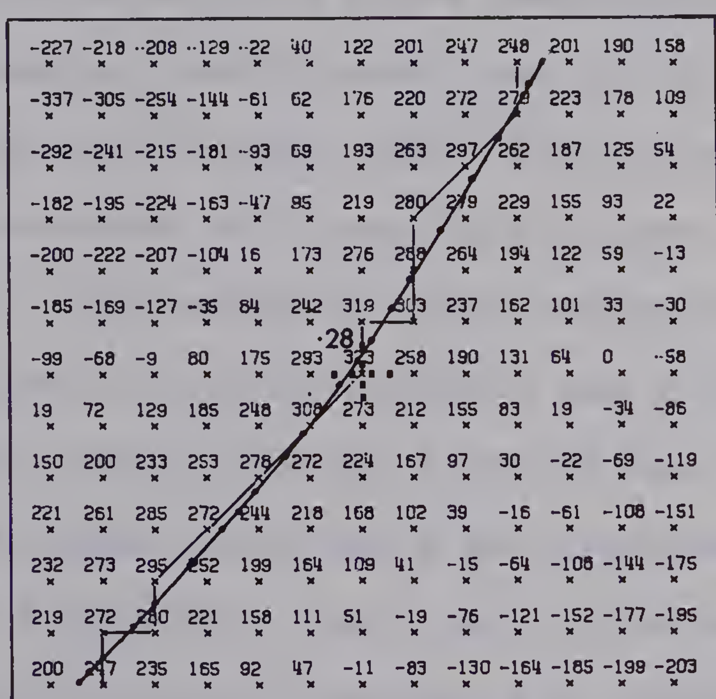
In Appendix F the flow chart for trend analysis indicates some of the logical decisions to be made by the computer in selecting the maximum coefficients and in determining the third degree polynomial equation for the trend line. The working Fortran IV programs IBM 67/360, for trend analysis are attached in Appendix F.



(a)
forward direction



(b)
Reverse direction



(c)
Smooth trend

Figure 3.6

(a) & (b) Scanning procedure to find the maxima coefficients for trend points

(c) Showing the smooth trend using third degree polynomial fit to the selected points (as obtained in figure 3.6(a) and (b).)

Examples from the field data

The method of trend analysis from the cross-correlation coefficients was applied to different sets of two-dimensional potential field data. In the first case the data was a matrix of 69 by 69 points obtained by digitizing the aeromagnetic map discussed in Chapter 2. The digitizing interval was ½ mile and the data was filtered and decimated to obtain a digital interval of one mile. Convolution was with a filter which cuts off all wavelengths shorter than 4 miles. A test map of 6 by 6 miles was selected for cross-correlation with the 18 by 18 miles sample map which provided the cross-correlation coefficients over 12 by 12 miles. For the entire matrix of data, sixty four correlation maps of the type illustrated in figure (3.6 c) were obtained on a Calcomp plotter. The method of trend analysis, as outlined above, was applied to each correlation set and the results were transformed to the original filtered map shown in figure (2.8), in which the primary and intermediate trends in the area are displayed. The 6 by 6 miles test map is too small to show discrimination between the larger trends and the secondary ones since the large scale features have dimensions much greater than 6 miles. To analyse the large scale primary trends, it is necessary either to take a larger test map or to decimate the filtered data to a 35 by 35 matrix size at 2 mile intervals (figure 3.8).

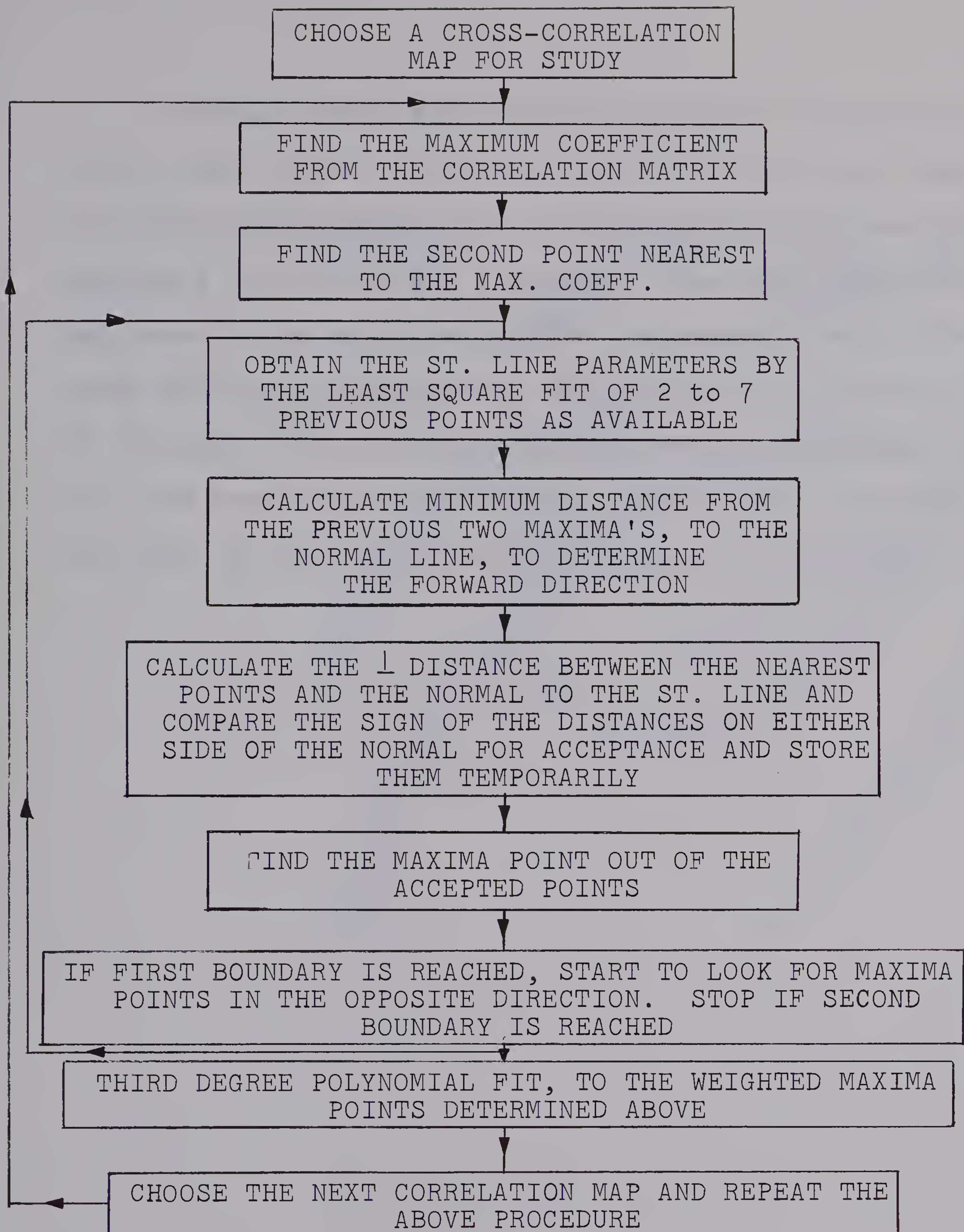


Figure 3.7. Block Diagram For Trend Analysis.

A similar study was carried out using the gravity data in the same region. An area of 52 by 52 miles was digitized at a one mile interval and was subjected to the same filter operators but it was not decimated. The test and sample maps are of the same size as for the magnetic data. The trend analysis using gravity data is shown in figure (2.9). It is clear from the study of the trend map that there are very few trends reflected by the gravity data and most of the gravity anomaly is due to the large norite mass.

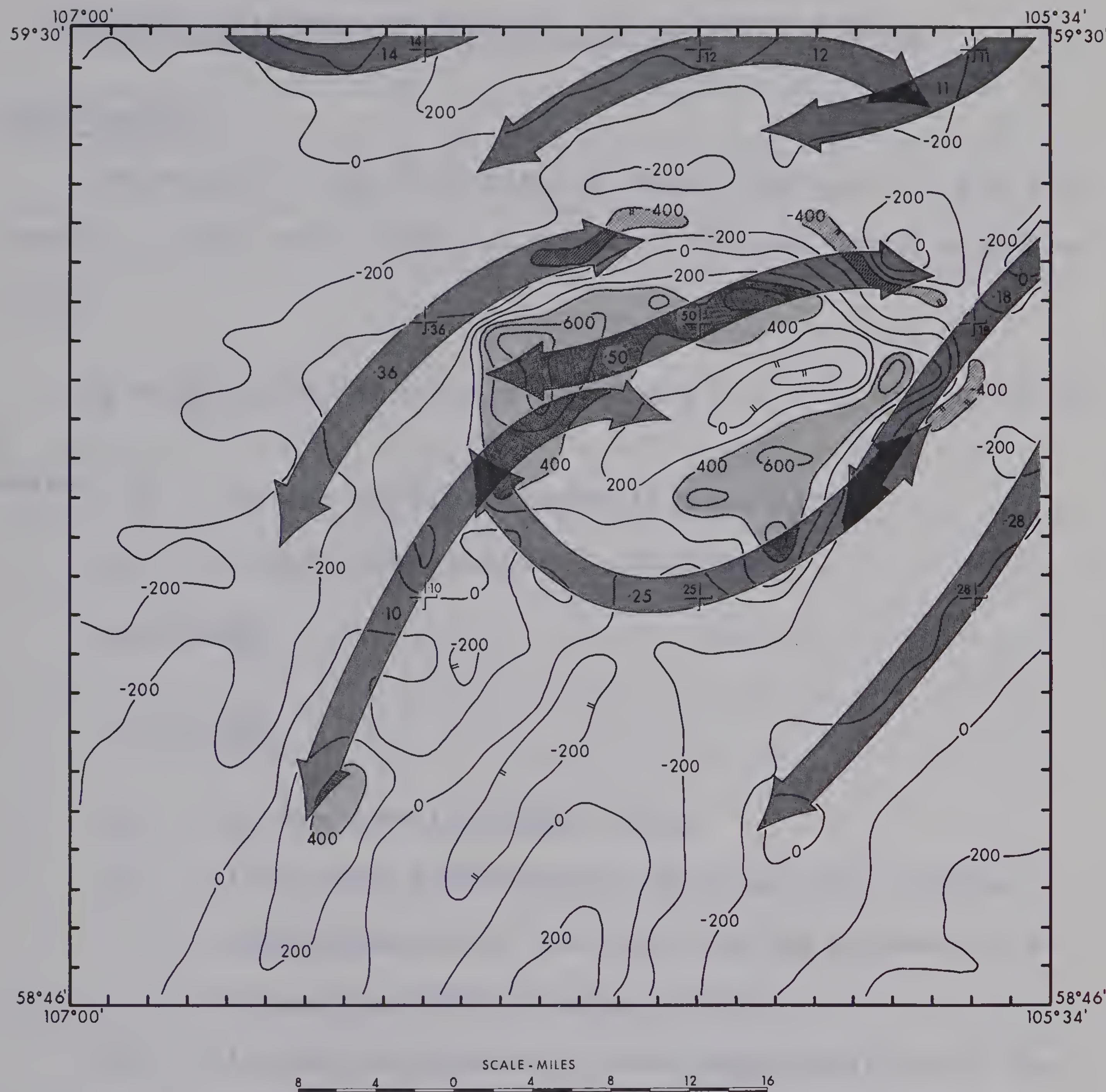


Figure 3.8

Showing the trends as obtained by using a test map of 12 by 12 miles for the magnetic data (decimated) in figure (2.8)

CHAPTER IV

MAGNETIC MAP REDUCED TO THE POLE AND COMBINED ANALYSIS OF GRAVITY AND MAGNETIC FIELDS

Introduction

The ratio of the intensity of magnetization, J, and the density of the rock units, ρ, can be obtained using equation (1.6).

$$Z_T = \frac{J}{G\rho}\,[g'(z)\,\sin I + g'(H)\,\cos I] \qquad \text{......(4.1)}$$

Where Z_T is the vertical magnetic intensity

G is the gravitational constant

$g'(z)=\frac{\partial g}{\partial z}$

$g'(H)=\frac{\partial g}{\partial H}$

g is the gravitational field

H is in the direction of declination D of the magnetization of the body and is assumed to be along the earth's normal field.

I is the inclination of the magnetization of the body and is assumed to be along the earth's normal field.

In this equation $g'(z)$ can be determined directly by using the continuation and derivative programs as described in Chapter 2. The horizontal gradient $g'(H)$ along the declination D is obtained by

$$g'(H) = g'(x)\,\text{Sin}\,D + g'(y)\,\cos D \qquad \text{......(4.2)}$$

Where

x is considered eastward and

y is considered northward

$g'(x)$ and $g'(y)$ are obtained by differentiating equation (1.1) with respect to x and y respectively. The Fortran IV program to obtain $g'(x)$ is listed in Appendix G. Figures 4.1 (a) and 4.1 (b) illustrate the first vertical derivative $g'(z)$ of gravity at 1000 feet above the surface and the first horizontal derivative $g'(H)$ along the declination 22°E, at the same level for the Stony Rapids area of Northern Saskatchewan. The original data discussed in Chapter 2 was filtered and decimated, so the sampling interval is 2 miles.

The theoretical component, Z_T, of magnetic intensity can be calculated using the observed gravity data, for any known or assumed direction of magnetization with the following restrictions on the sources:

(1) The gravitational and magnetic fields are produced by the same body and

(2) the body has a uniform density, ρ, and is polarized uniformly with an intensity of magnetization J.

Figure (4.2) illustrates the theoretical component of magnetic intensity, Z_T, in which the ratio $\frac{J}{\rho}$ is assumed to be constant over the entire area.

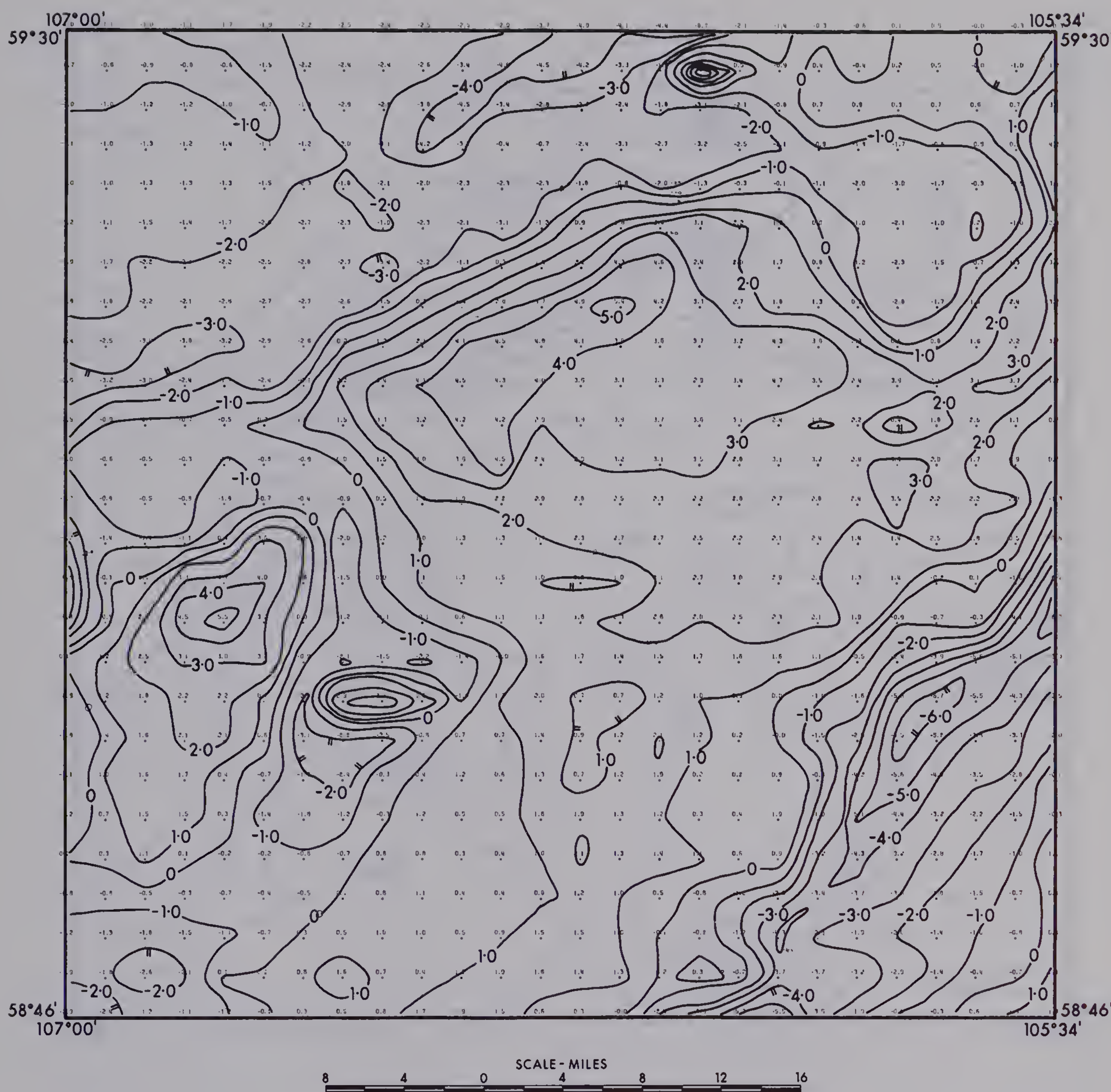


Figure 4.1 (a)

First vertical derivative, g′(z) of gravity field at 1000 feet above the surface

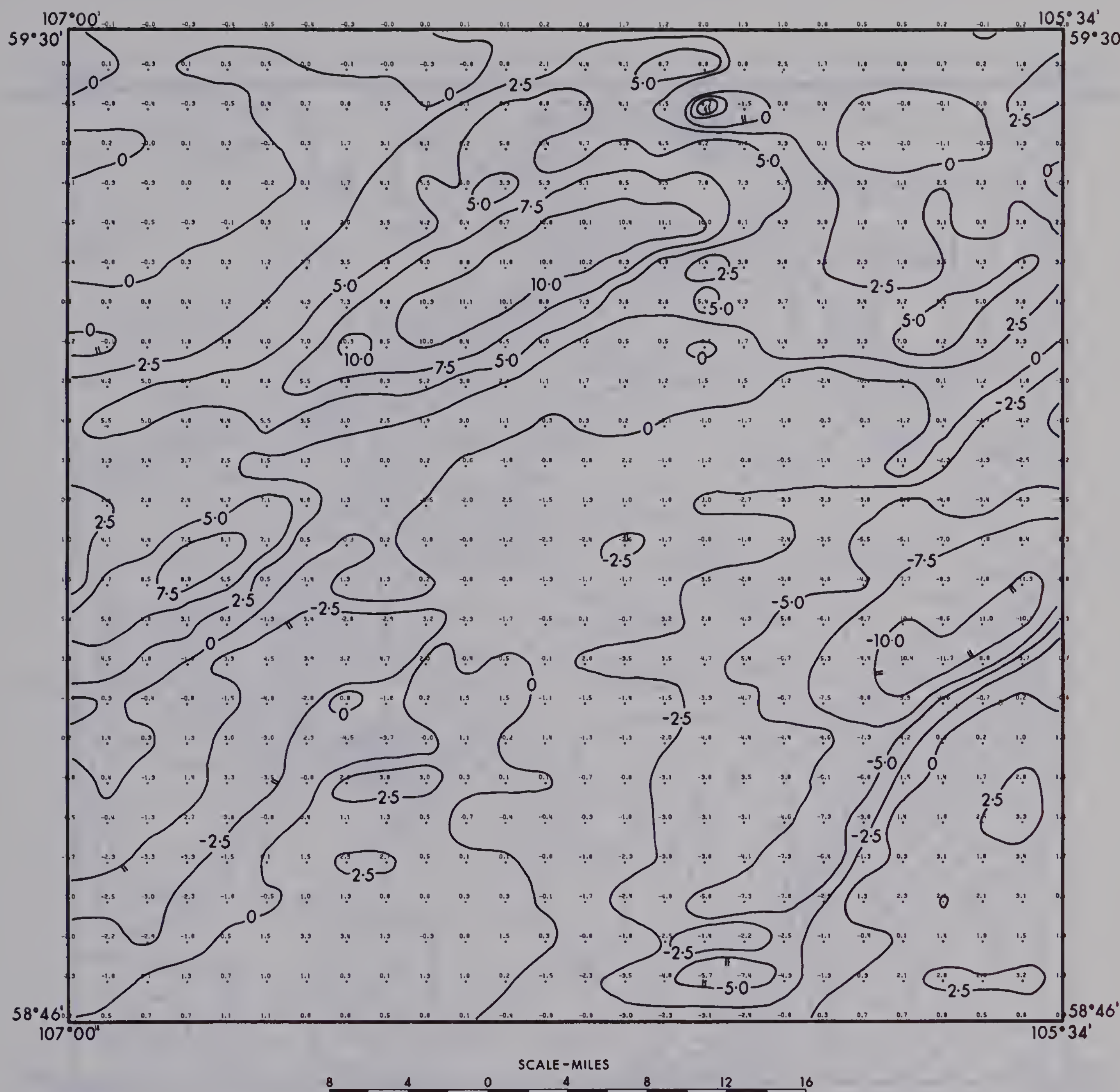


Figure 4.1 (b)

First horizontal derivative, $g'(H)$ along the declination 22° E

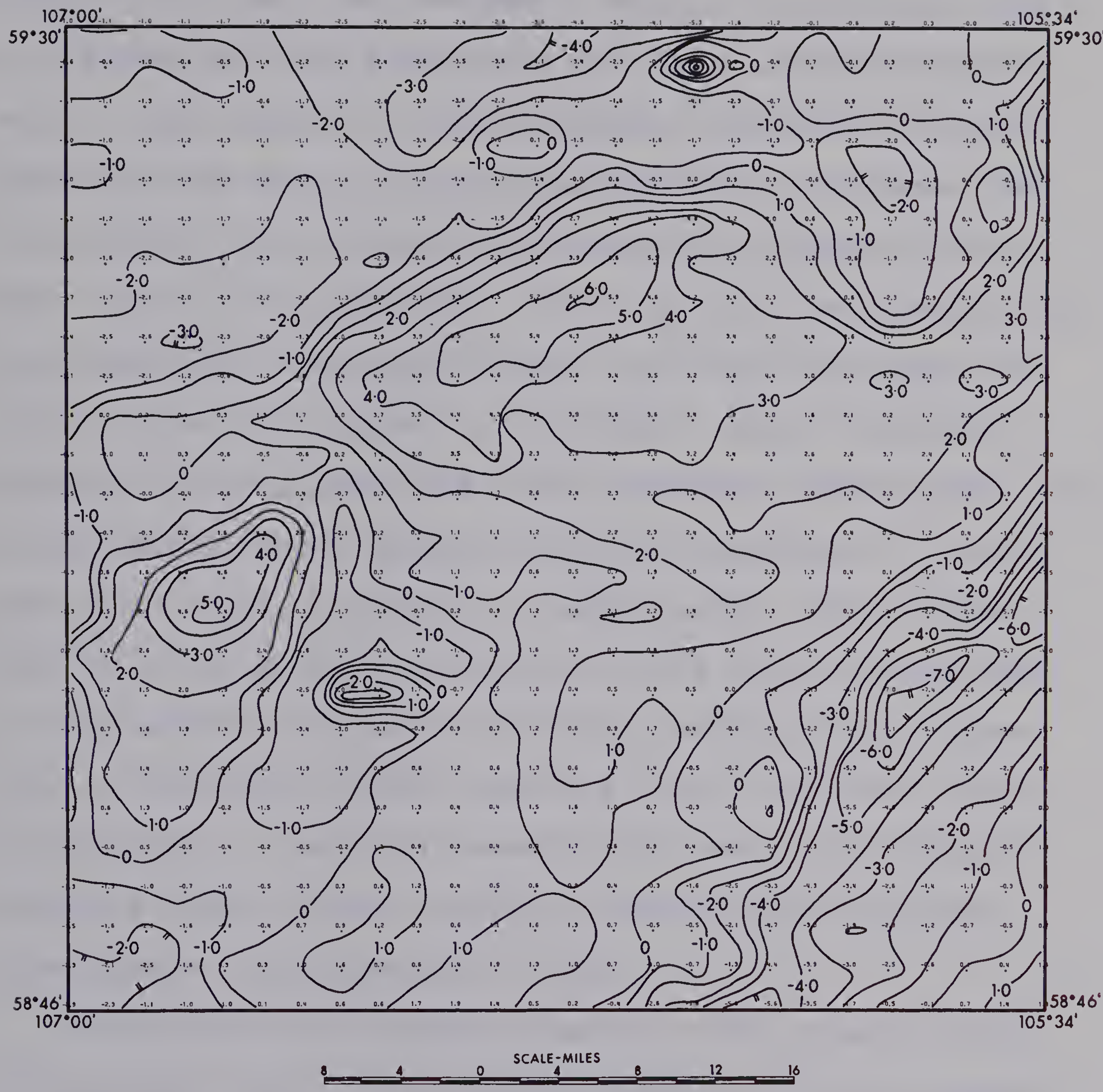


Figure 4.2

Theoretical component of vertical magnetic intensity Z_T equals plotted values × J/(ρG)

General Geology, gravity and magnetics of the Stony Rapids area

The general geology of the area is briefly reviewed from the published geological works mentioned below. North of the Fond-du-Lac river, meta-sediments make up the dominant rocks of the area and also associated with them are meta-volcanic rocks. They consist of biotite-gneiss, hornblende-biotite-gneisses, meta-arkose, garnet-quartz-feldspar gneisses, etc. The granitic rocks consist of granodiorite, pegmatite and minor granitic segregations. Their outcrops are numerous and form pronounced topographic highs. In the study area, the noritic rocks are exposed in the north. Small concordant bodies of noritic rocks are found intimately interlayered with the garnet-gneisses, arkoses and biotite-gneisses. Diabase dikes also occur, at places, in small sizes. For further details of the geological mapping in the area, one may refer to the reports by Colborne (1960-63), Fahrig (1961), Blake (1956), Hriskevich (1949), Mawdsley (1949), Furnival (1940), Alcock (1936). South of Fond-du-Lac river, the Athabasca sandstone crops out and nothing is known about the underlying igneous and metamorphic rocks.

From gravity and magnetic maps it seems apparent that the potential field anomalies are caused by the high density mass distribution underneath (e.g. norite bodies).

A negative gravity anomaly trend, of about -10 to -15 milligals runs on either side of the main gravity anomalies and are in the NE-SW direction. Observations from the magnetic map show the low magnetic trends in the same general directions as the negative gravity anomaly trends and these are large scale trends which continue across into Alberta. Most probably, these large scale trends are associated with long fractures or shear zones which may be zones of weakness in the crust. A NE-SW trending fault in the northwest of the main intrusive body coincides with the low magnetic and gravity anomaly trends. Furthermore, the low gravity and magnetic trends southwest of the main intrusive body seems to parallel the Black Lake fault.

The vertical component of gravity was calculated for a three-dimensional model using the density differences described by Agarwal and Kanasewich (1968). Figure (4.3 a) shows a cross-section through the center of this model and the observed values along the line 6-6'. Good agreement was obtained between Bouguer anomalies and the model gravity calculations. The theoretical profile was calculated by using the method devised by Talwani and Ewing (1960). The model chosen for analysis is not unique as there could be numerous solutions for various distributions of the same total mass. The model was divided into two parts with a density contrast of 0.25 gm/cm^3 between basic and country rocks and -0.3 gm/cm^3

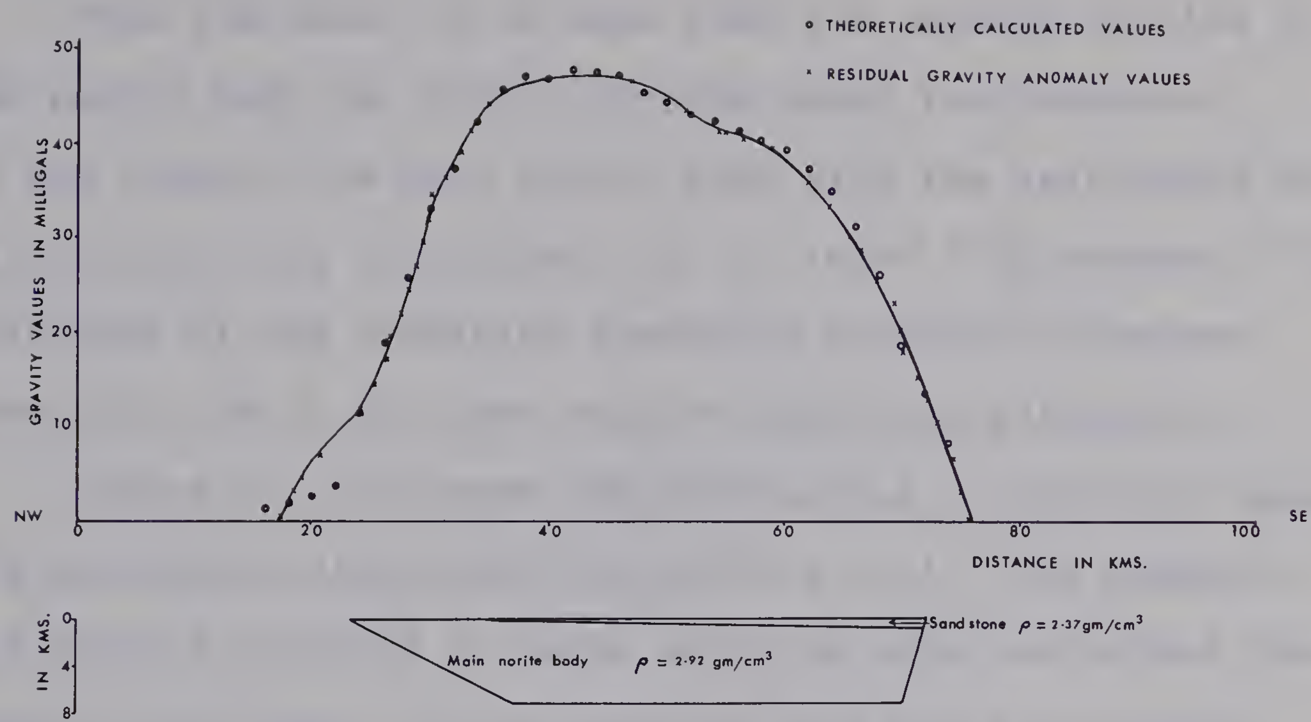


Figure 4.3 (a)

Theoretical model of gravity profile 6-6' shown in figure (2.1)

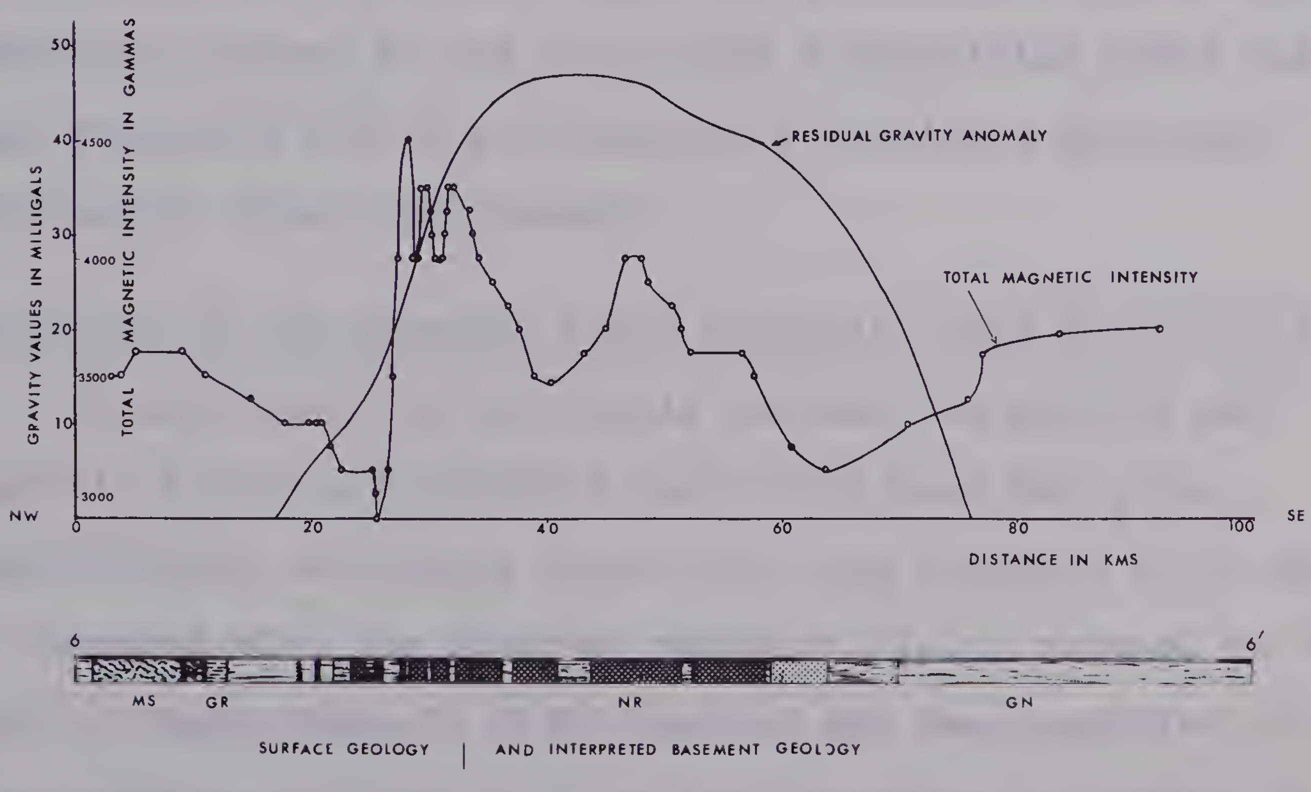


Figure 4.3 (b)

Correlation of gravity, magnetic and geological data along the profile 6-6'

between sandstone and Country rocks.

From the model it is seen that the exposed portion of the norite body is gently dipping under the sandstone. In the centre, the main norite body with the calculated mass distribution has a thickness of at least 7 kilometers. The thickness of the overlying sandstone probably increases gradually, NW to SE, from zero to about one kilometer.

Figure 4.3 (b) shows the correlation of gravity, magnetic and geological data along the profile 6-6'. The magnetic and gravity profiles do agree with the main geological features in the area. It is observed that there are interrelated gneisses in the norite body, which are not taken into account in our simple model. The model presents a general distribution of anomalous mass with a moderate thickness of sandstone blanket at the top. Only a simplified model has been presented and it will need to be modified with more information about the basement.

Reduction of the observed total magnetic field to the pole

To establish the hypothesis whether the gravity and magnetic fields are produced due to the same body, the theoretically calculated value, Z_T, from equation (4.1) must be compared with the observed magnetic field 'reduced to the pole'. Bhattachayya's (1965) method has been modified to take advantage of the fast Fourier transforms to reduce the total magnetic field to the pole for arbitrary values of

inclination and declination. The formula used and the Fortran IV program is given in Appendix H. Figure (4.4) illustrates the data, reduced to the pole, using an inclination of 80°N and a declination of 22°E. The figures are the current values in the Stony Rapids area as obtained from the maps published by the Dominion Observatory.

Analysis of gravity and magnetic fields in the wave number domain

The potential field presents the composite effect due to sources of various wavelengths. One can analyze each wavelength in the wave number domain by taking the Fourier transforms of each set of data and it may be possible to arrive at some suitable idea about the type of source, a particular wavelength may represent. The typical characteristic physical properties, such as $\frac{J}{\rho}$, for a particular band of wavelengths may provide an insight about the various rock unit structures and could be helpful in the interpretation of potential field data.

The two sets of data illustrated in figures (4.2) and (4.4) can be analyzed by using the coherency criteria which is a function of wave number. The coherence function is defined as:

$$\mathrm{Coh}^2_{Z_o,Z_T}(k_x,k_y) = \frac{\left|P_{Z_o,Z_T}(k_x,k_y)\right|^2}{P_{Z_o}(k_x,k_y)\;P_{Z_T}(k_x,k_y)} \qquad \ldots\ldots(4.3)$$

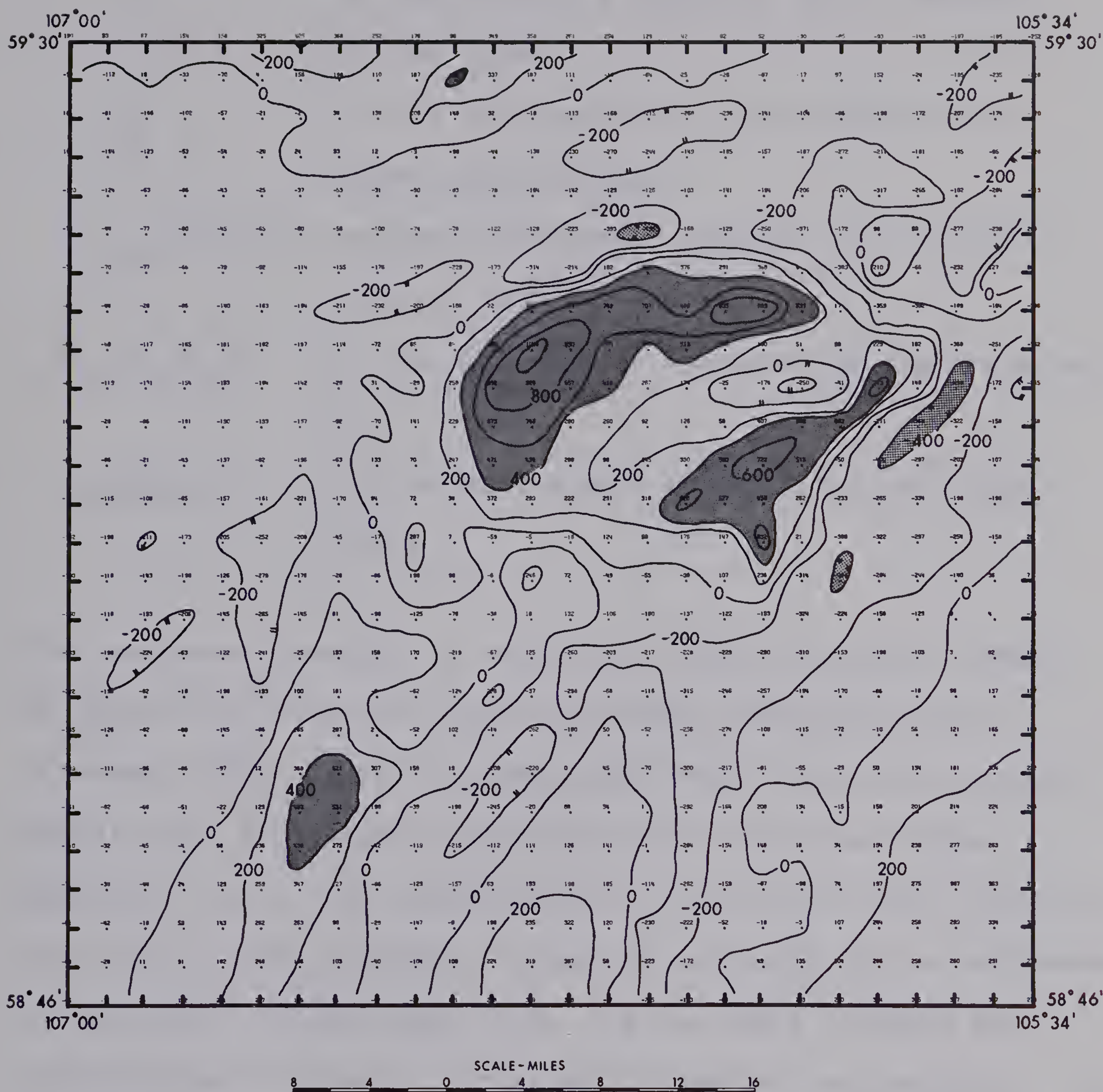
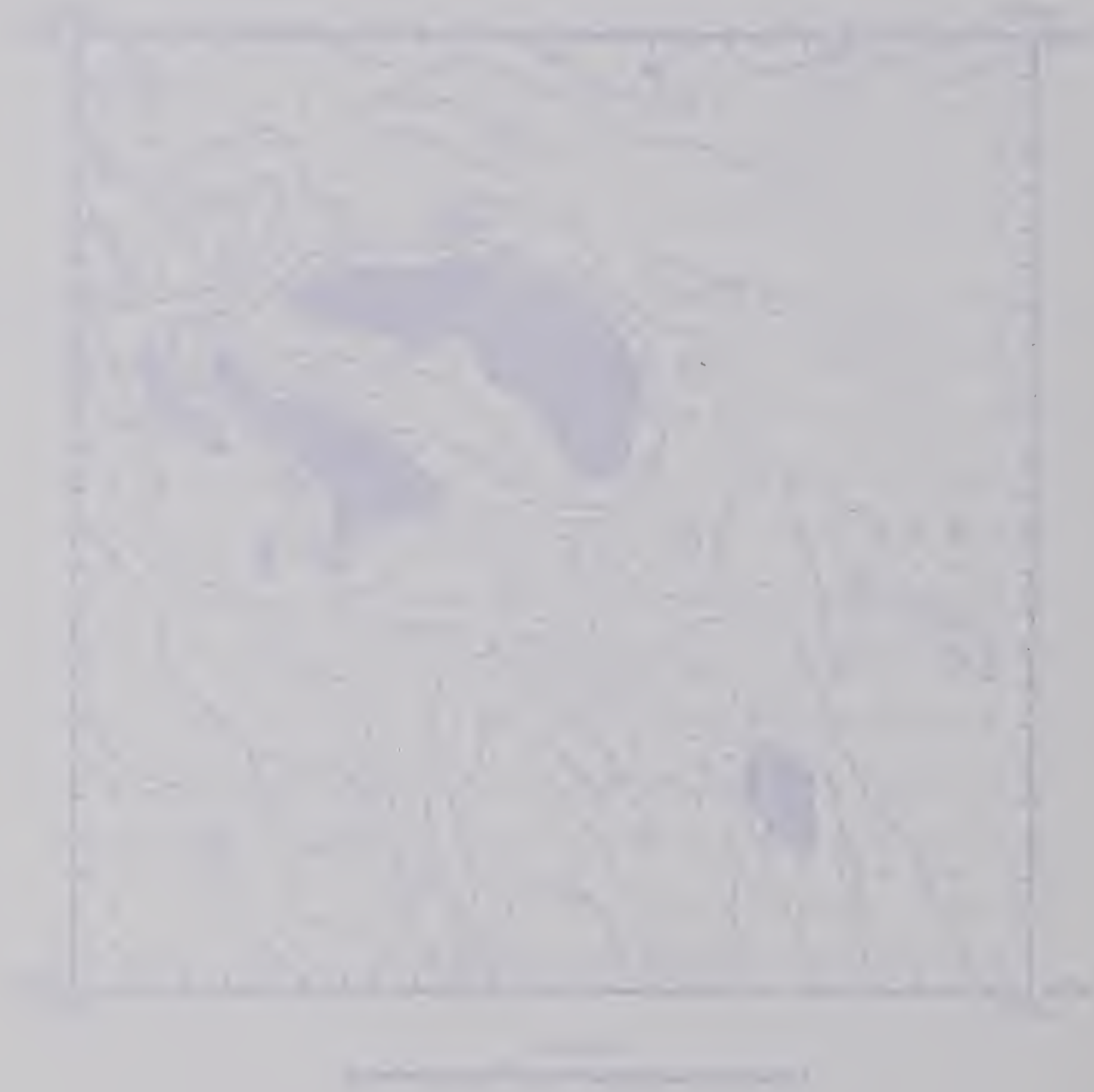


Figure 4.4

Total magnetic field (figure 3.11) 'reduced to the pole' for $I=I_0=80^\circ$ N and $D=D_0=22^\circ$ E values for the earths' normal field.



Where

Z_o	is the observed magnetic data 'reduced to the pole'.
Z_T	is the theoretically calculated field using gravity data.
$P_{Z_o,Z_T}(k_x,k_y)$	is the cross-power spectra for the two sets of data Z_T and Z_o.
$P_{Z_o}(k_x,k_y)$	is the auto-power spectra for the data set Z_o.
$P_{Z_T}(k_x,k_y)$	is the auto-power spectra for the data set Z_T.

The raw power density is smoothed in the frequency domain by convolving with the two-dimensional Hamming window (Kinsman 1965). From the smoothed cross and auto-spectral density the coherence coefficients are obtained using equation (4.3). If the coherency is very low for a particular band of wave numbers, Z_o and Z_T are said to be incoherent in that wave number band, i.e. the two sets of data are uncorrelated in phase over a small band of wavelengths. It will clearly indicate that the magnetic and gravity data do not represent or are not due to the same source body and the method to obtain $\frac{J}{\rho}$ in such a case will not be applicable. It may also indicate that the rocks are magnetized in a different direction from the earth's normal field. On the other hand, if the coherency is very high (Coh $\geq$ 0.7) for

a particular band of wave numbers, then the source of the magnetic and gravity fields are probably from the same rock units. In such a case the method would provide a good set of $\frac{J}{\rho}$ values which can be calculated from the amplitude of each data set.

Knowing that the two sets of data are highly coherent, one can proceed to obtain the $\frac{J}{\rho}$ ratio for a particular band of wave numbers. This may be done by taking the Fourier transforms of each set of data separately and then calculating the amplitude for each data set. The amplitude ratio from the observed magnetic field to the theoretically calculated one would provide the representative value of $\frac{J}{\rho}$ for a particular band of wavelengths. For a highly coherent potential fields, the values of $\frac{J}{\rho}$ should be uniform.

Any inconsistency of $\frac{J}{\rho}$ or coherency tests would reflect whether the underlying assumptions such as the direction of magnetization and the uniform properties of the source body are valid or not. If the results are not coherent, Z_T can be re-calculated from equation (4.1) by assuming different directions of magnetization and carrying over the process all over again. For a more complete analysis equation (4.1) will have to be generalized to the case where the direction of magnetization of the rocks and of the earth's normal field are different.

If the method is carried out to completion, it will supply not only $\frac{J}{\rho}$ ratio, representative of rock units in the

particular area, but one can also obtain J knowing ρ from rock sample measurements or vice versa. The iteration of the method would also provide the direction of magnetization.

Application to potential field data

Coherency test for the two sets of data as illustrated in figures (4.2) and (4.4) is plotted in figure (4.5) and the $\frac{J}{\rho}$ ratio for various wavelengths are shown in figure (4.6).

Generally, the coherency is high and the $\frac{J}{\rho}$ ratio in the study area varies between 1000×10^{-6} cgs units to $20{,}000 \times 10^{-6}$ cgs units for 3 to 14 mile long wavelengths. Assuming a density contrast of 0.25 gm/cm^3 the calculated susceptibility varies from 2000×10^{-6} cgs units to 8000×10^{-6} cgs units using the 0.6 Oersted field in this area. The susceptibility values, thus obtained agrees generally with the basic type of rocks given by Garland (1951). It is reasonable to assume from this analysis that both the density contrast and the magnetization is due to the norite and associated basic rocks in the Stony Rapids area.

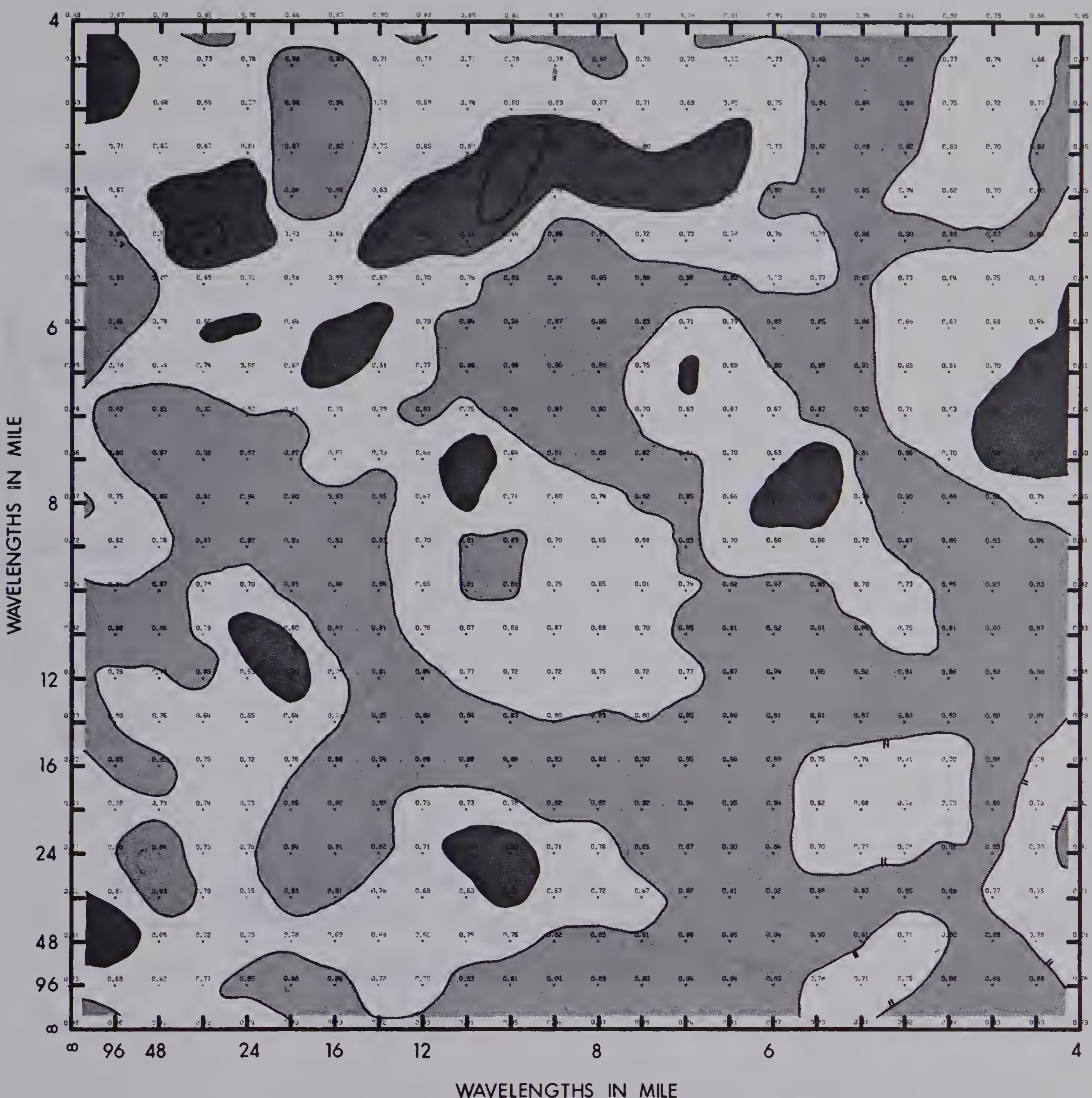


Figure 4.5

Coherency values for various wavelengths (using the 'reduced to the pole' and the calculated data).

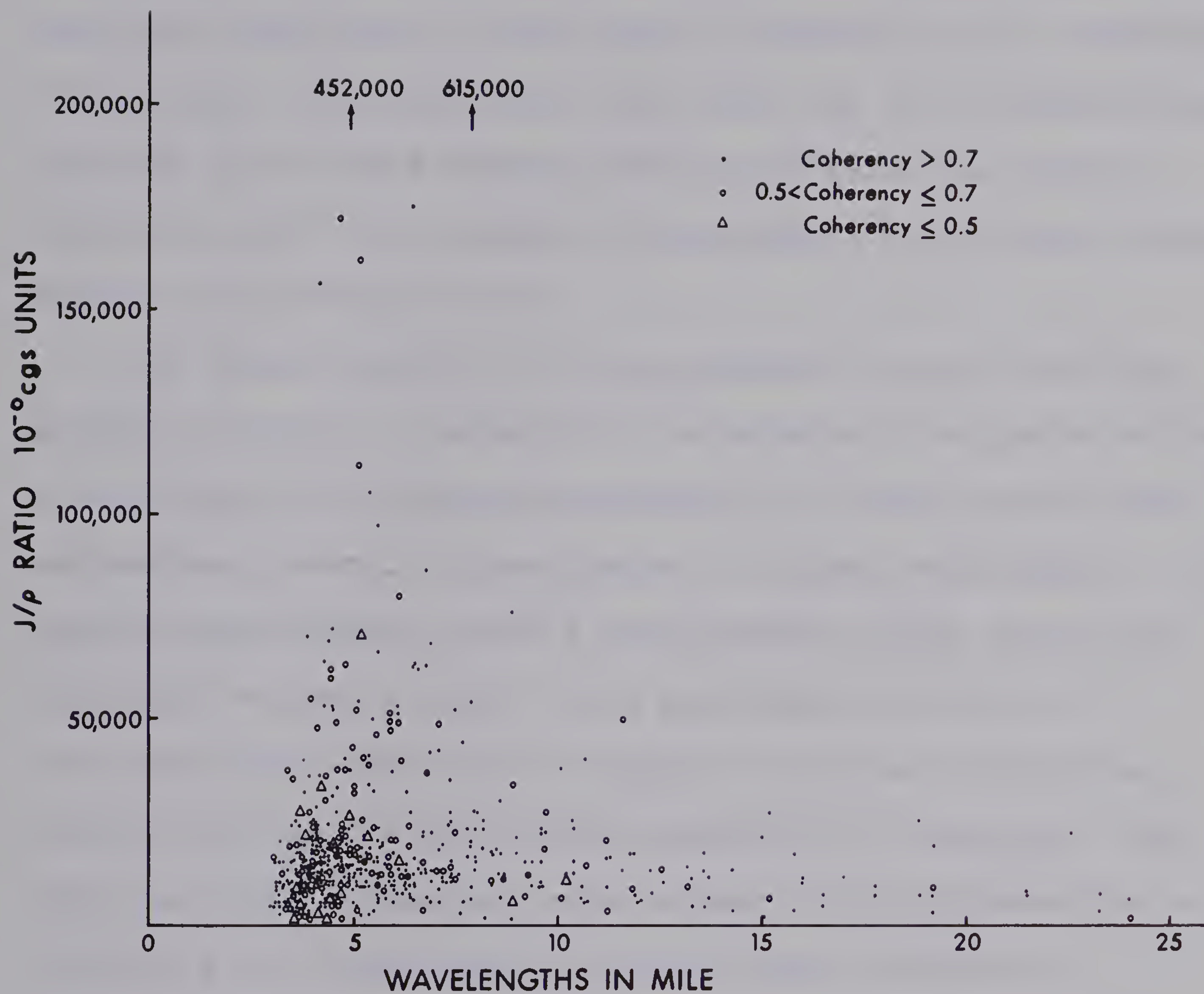


Figure 4.6

Ratio's of J/ρ for various bands of wavelengths in the Stony Rapids area, Northern Saskatchewan.

CONCLUSIONS

This investigation is concerned with the two-dimensional harmonic analysis in the digital domain using recently developed techniques in time sequence. Use of the fast Fourier transform algorithm has been made in developing the techniques of analysis. The potential field data can be represented and analysed in the wave number domain to obtain the physical characteristics for a group of wavelengths, which may further enhance the interpretation.

The power spectra of a long magnetic profile has been studied in detail to establish the effects of aliasing errors on structures of various wavelengths. It was observed that the maximum power is concentrated at longer wavelengths. At shorter wavelengths, quite a few spectral highs were noted which may be due to small scale geological features or represents the noise due to digital errors and should be filtered out before any further analysis of the data. This study has also indicated there is very little distortion or aliasing at all wavelengths greater than 1 mile for a sampling interval of ½ mile. Probably a sampling interval of 1 mile will suffice if one is interested in studying the structures longer than 4 mile wavelengths. A digital interval as large as 2 miles is dangerous to use because a substantial amount of distortion is introduced at longer wavelengths in this study area. On the other hand, to study the small scale structures of economic significance, a smaller digital interval may be desirable.

The study was limited to large scale structures and with this in mind, the magnetic maps were digitized at ½ mile intervals obtaining 20,000 points. After convolution and decimation, the digitized data was further reduced to 5000 points at a sampling interval of 1 mile. The impulse response and transfer functions for the two-dimensional filters, used for smoothing the data, were studied. It is concluded that these filter operators are quite adequate although they could be further improved if desired. A study of the maps indicates that the filtering process could be very useful in delineating the trends for various wavelengths of interest. Particularly, the filtered maps using band pass (2-20 miles) operators delineates the trends very effectively.

During this work a new method of cross-correlation has been developed from which trend directions can be obtained with more reliability. In the new method of cross-correlation the same two dimensional matrix of values are used for the two functions but one of them is modified by surrounding the central core with a field of zero's. This important concept of introducing zero's around the test function makes it possible to obtain the correlation coefficients for the same sample set without the limitations imposed by the symmetry properties of an auto-correlation function. A highly correlated data set provides a high correlation coefficient

and an increase or decrease of the coefficient value is an indication of trend direction. This technique makes it possible to study the primary or secondary or both types of trends, although the present study is limited to primary trends only. To study the large scale features of interest, the two-dimensional data was convolved with the filter operators to remove the features due to short wavelengths and then decimated to reduce the number of coefficient calculations. The direct method of calculating the cross-correlation coefficients takes an enormous amount of computer time which prevents one from undertaking a study on a large scale. To overcome this difficulty a method has been developed in which the calculations of two-dimensional cross-correlation coefficients are carried out by using the fast Fourier transform algorithm.

The cross-correlation coefficients are normalized so they vary between +1 and -1 and their spatial variation provides a method of tracing the trends due to various geological factors. An empirical method has been devised to trace the trends in an unbiased manner by fitting a least squares third order polynomial to the maximum trend coefficients. The technique has been applied to the magnetic and gravity data from the Stony Rapids area, Northern Saskatchewan. The method provides a weighted correlation coefficient for each trend line which can be used as a criterion in accepting or

rejecting a particular trend. It is possible to design a filter which may remove the effect of a particular trend and allow the unbiased analysis of local anomalies of interest.

In another aspect of this investigation, analytic techniques such as upward and downward continuation, first, second etc. vertical derivatives, first horizontal derivatives and reduction of the total magnetic field to the pole have been developed by using the fast Fourier transform algorithm. This simplifies the calculations and provides the results in a compact form. Use of these techniques have been made in the combined analysis of magnetic and gravity data. The possibility of calculating representative $\frac{J}{\rho}$ ratios of the rock units in a particular area is investigated. It is done by analysing each wavelength, of the potential field data, in the wave number domain. A coherency criterion is used to test the validity of two potential field data sets for such an analysis. The method is limited to a particular case where the rocks are magnetized in the direction of the earth's normal field. If the coherency criterion is valid, one can obtain a representative set of $\frac{J}{\rho}$ ratio as a function of wavelength in a particular area and this could be useful in the interpretation of the potential field data.

BIBLIOGRAPHY

Affleck, J., 1963, "Magnetic anomaly trend and spacing patterns", Geophysics, Vol. 28, no. 3, pp. 379-395

Agarwal, R.G. and E.R. Kanasewich, 1968, "A gravity investigation of the Stony Rapids area, Northern Saskatchewan", Sask. Dept. of Min. Res. (to be published)

Alcock, F.J., 1936, "Geology of the Lake Athabasca region, Saskatchewan", Geol. Survey of Canada, mem.196

Alldredge, L.R. and G.D. Van Voorhis, 1961, "Depth to sources of magnetic anomalies", Jour. of Geophys. Res. Vol.66, pp. 3793-3800

Alldredge, L.R., G.D. Van Voorhis and T.M. Davis, 1963, "A magnetic profile around the world", Jour. of Geophys. Res., Vol. 68, pp. 3679-3692

Bacon, R.H., 1953, "The "Best" straight line among the points", American Jour. of Physics, Vol.21, no. 6, pp. 428-446

Baranov, V., 1957, "A new method for interpretation of aeromagnetic maps: Pseudo Gravimetric anomalies", Geophysics, Vol. 22, no. 2, pp. 359-383

Baranov, V. and H. Naudy, 1964, "Numerical calculation of the formula of reduction to the magnetic pole", Geophysics, Vol. 29, no. 1, pp. 67-79

Bendat, J.S. and A.G. Piersol, 1966, "Measurement and Analysis of random data", (published by John Wiley and Sons)

Bhattacharyya, B.K., 1965, "Two-dimensional harmonic analysis as a tool for magnetic interpretation", Geophysics, Vol. 30, no. 5, pp. 829-857

Bhattacharyya, B.K. and L.W. Morley, 1965, "The delineation of deep crustal magnetic bodies from total field aeromagnetic anomalies", Jour. of Geomagnetism and Geoelectricity, Vol. 17, no. 3-4, pp. 237-252

Bhattacharyya, B.K., 1966, "Continuous spectrum of the total magnetic field anomaly due to a rectangular prismatic body", Geophysics, Vol. 31, no. 1, pp. 97-121

Blackman, R.B., and J.W. Tukey, 1958, "The measurement of power spectra", (Dover publication)

Blackman, R.B., 1965, "Linear data smoothing and prediction in theory and practice", (published by Addison-Wesley publishing Co.)

Blake, D.A.W., 1956, "Geological notes on the region south of Lake Athabasca and Black Lake, Saskatchewan and Alberta", Geol. Survey of Canada, paper 55-33

Bracewell, R., 1965, "The Fourier transform and its applications", (McGraw-Hill Book Company)

Bullard, E.C. and R.I.B. Cooper, 1948, "The determination of the masses necessary to produce a given gravitational field", proc. Royal Soc. of London, Vol. 194A, pp. 332-347

Byerly, P.E., 1965, "Convolution filtering of gravity and magnetic maps", Geophysics, Vol 30, no. 2, pp. 281-283

Chapman, S. and J. Bartels, 1940, Geomagnetism, Vol. 1 and 2 (Oxford Press)

Colborne, G.L., 1960, "The geology of the Clut Lakes area (west half) Saskatchewan", Sask. Dept. of Mineral Resources, Report no. 43

Colborne, G.L., 1961, "The geology of the Clut Lakes area (east half) Saskatchewan", Sask. Dept. of Mineral Resources, Report no. 58

Colborne, G. L., 1962, "The geology of the Wiley Lake area (east half) Saskatchewan", Sask. Dept. of Mineral Resources, Report no. 69

Colborne, G.L. and E. Rosenberger, 1963, "The geology of the Wiley Lake area (west half) Saskatchewan", Sask. Dept. of Mineral Resources, Report no. 79

Cooley, J. W. and J. W. Tukey, 1965, "An algorithm for the machine calculation of complex Fourier series", Math. of computation, Vol. 19, no. 90, pp. 297-301

Cooley, J.W., P.A.W. Lewis and P.W. Welch, 1967, "The fast Fourier transform algorithm and its applications", IBM Research paper RC-1743, pp. 1-157

Dean, W.C., 1958, "Frequency analysis for gravity and magnetic interpretation", Geophysics, Vol. 23, no. 1, pp. 97-127

Deming, W.E., 1943, "Statistical Adjustment of Data", (John Wiley & Sons)

Fahrig, W.F., 1961, "The geology of the Athabasca formation", Geol. Survey of Canada, Bull. 68

Fuller, B.D., 1966, "Two dimensional frequency analysis of grid operators", presented at the 36th meeting of SEG in Houston, Texas, U.S.A.

Furnival, G.M., 1940, "Stony Rapids and Porcupine River areas, Saskatchewan", Geol. Survey of Canada, paper 40-10

GAG Report No. 9, 1955, Geophysical analysis group, Dept. of Geology and Geophysics, MIT, Cambridge, U.S.A.

Garland, G.D., 1951, "Combined analysis of gravity and magnetic anomalies", Geophysics, Vol. 16, no. 1, pp. 51-62

Gentleman, W.M. and G. Sande, 1966, "Fast Fourier transforms-for fun and profit", proc. of the fall joint computer conference, San Francisco, California, pp. 563-578

Gladkii, K.V., 1960, "Frequency analysis in the processing and interpretation of results of Gravitational and Magnetic Surveys, and the possibility of the automatic solution of the processing and interpretation problem", Trans (Trudy) MINKH and GP, no. 31

Goldstein, N.E., 1962, "Numerical filtering of potential field signals as applied to geophysical exploration", M.Sc. Thesis, Univ. of California

Good, I.J., 1958, "The interaction algorithm and practical Fourier analysis", J.R. Statist. Soc., ser. B. Vol 20, pp. 361-372

Good, I.J., 1960, "An addendum, Vol. 22, pp. 372-375

Granger, C.W.J. in association with M. Hatanaka, 1964, Spectral analysis of economic time series", (Princeton Univ. Press)

Gudmundsson, G., 1966, "Interpretation of one dimensional magnetic anomalies by use of the Fourier transform", The Geophysical Jour. Vol. 12, no. 1, pp. 87-98

Hahn, A., 1965, "Two applications of Fourier analysis for the interpretation of geomagnetic anomalies", Jour. of Geomagnetism and Geoelectricity, Vol. 17, no. 3-4, pp. 195-225

Hall, D.H., 1964, "Magnetic and tectonic regionalization on Texada Island, British Columbia", Geophysics, Vol. 29, no. 4, pp. 565-581

Horton, C.W., W.B. Hempkins and A.A.J. Hoffman, 1964, "A statistical analysis of some aeromagnetic maps from

the Northwestern Canadian Shield", Geophysics, Vol. 24, no. 4, pp. 582-601

Hriskevich, M.E., 1949, "Radioactive occurrences in the Black Lake area, Athabasca Mining Division, Saskatchewan", Sask. Dept. of Natural Resources, Precambrian Geology Ser. Report 2

Kanasewich, E.R., 1966, Lecture notes on "Time sequence analysis in the Earth Sciences", Course physics 623

Kinsman, B., 1965, "Wind Waves", p. 466 (Prentice Hall)

Lee, Y.W., 1960, "Statistical Theory of Communication", (John Wiley & Sons)

Mawdsley, J.B., 1949, "Pine Channel area, Lake Athabasca district, Saskatchewan", Geol. Survey of Canada, paper 49-27

McClure, D., 1963, "Interpretation of long aeromagnetic profiles", M.Sc. Thesis, Univ. of Alberta, Edmonton, Alberta

Mesko, A., 1965, "Some notes concerning the frequency analysis for gravity interpretation", Geophysical prospecting, Vol. 13, pp. 475-488

Odegard, M.E. and J.W. Berg, Jr., 1965, "Gravity Interpretation using the Fourier Integral", Geophysics, Vol. 30, no. 3, pp. 424-438

Papoulis, A., 1962, "The Fourier Integral and its applications", (McGraw-Hill Book Company)

Parratt, L.G., 1961, "Probability and Experimental Errors in Science", An Elementary Survey, (John Wiley & Sons)

Robinson, E.A., 1966, "Collection of Fortran II programs for filtering and spectral analysis of single channel time series", Geophysical prospecting, Vol. 14, no. 1

Runge, C., 1903, Zeit. fur Math. und Physik, Vol. 48, p.443

Runge, C., 1905, Zeit. fur Math. und Physik, Vol. 53, p.117

Serson, P.H. and W.L.W. Hannaford, 1957, "A statistical analysis of magnetic profiles", Jour. of Geophys. Res. Vol. 62, no. 1, pp. 1-18

Singleton, R.S., 1967, "on computing the fast Fourier transform", Comm. of the Assos. for Comp. Mach., Vol. 10, no. 10, pp. 647-654

Spector, A. and B.K. Bhattacharyya, 1966, "Energy density spectrum and auto-correlation function of anomalies due to simple magnetic models", Geophysical prospecting, Vol. 14, no. 3, pp. 242-272

Swartz, C.A., 1954, "Some geometrical properties of residual maps", Geophysics, Vol. 19, no. 1, pp. 46-70

Talwani, M. and M. Ewing, 1960, "Rapid computations of gravitational attraction of three-dimensional bodies of arbitrary shape", Geophysics, pp. 204-225

Tsuboi, C. and F. Fuchida, 1938, "Relations between gravity values and corresponding subterranean mass distribution", Bull. Earthquake Res. Inst. Tokyo, Vol. 15, pp. 636-649

Tsuboi, C., 1959, "Application of double Fourier series to computing gravity anomalies and other gravimetric quantities at higher elevations, starting from given surface gravity anomalies", Inst. of Geodesy, photography and cartography, Ohio State Univ. Report no. 2

Vestine, E.H. and N. Davids, 1945, "Analysis and interpretation of geomagnetic anomalies", Terres. Magnetism and Atmos. Electricity, Vol. 50, pp. 1-36

Wiener, N., 1949, "Extrapolation, Interpolation and Smoothing of stationary time series", (MIT Press)

Wilkinson, J.H., 1967, "The solution of ill-conditioned linear equations", in Math. Methods for Digital Computers edited by Ralston, A. and H.S. Wilf, pp.65-93 (John Wiley & Sons)

APPENDIX A

The fast Fourier transforms

The criteria for the formulation of equations (2.1) and (2.2) along with their various properties have been dealt thoroughly by Cooley and Tukey (1965), Gentleman and Sande (1966) and Cooley et al (1967). In general, direct computations of the transforms require N^2 operations, whereas by using the fast Fourier transform algorithm it takes less than $rN \log_r N$ operations (where r is a factor of N). In other words, one can obtain the transforms $(N^2/rN \log_r N)$ times faster, moreover the accuracy of the results is also increased by the same ratio. In the F.F.T. algorithm the data is considered in the form of a vector and the fundamental basis of the algorithm is to decompose each vector into simple factors. For example if N has r factors say $N = r_1 \cdot r_2 \ldots\ldots r_m$, then according to the F.F.T. algorithm a vector is decomposed into elementary transformations followed by a permutation of the result. In the same manner a two-dimensional data set of N_1 by N_2 size is considered to be a vector of length $N = N_1 N_2$ and the fast Fourier algorithm applied in the usual way. The resulting coefficients giving the amplitude of each harmonic are given by an array of N_1 by N_2 points.

Power spectra

For a one-dimensional set of data X_t, the Fourier transforms F_k are given by

$$F_k = \sum_{t=0}^{N-1} X_t \, e^{\frac{2\pi ikt}{N}} \qquad \text{for} \quad k=0,1,\ldots,(N-1)$$

......(A-1)

Using Equation (A-1), we can obtain the auto-correlations A_t in the following manner.

$$A_t = \frac{1}{N'} \sum_{k=0}^{N'-1} \left|F_k\right|^2 e^{\frac{-2\pi ikt}{N'}} \qquad \text{for} \quad t=0,1,\ldots,(N'-1)$$

......(A-2)

Where

$$F_k = 0 \quad \text{for} \quad k = N+1, N+2, \ldots N' \quad \} \quad N' > N$$

Thus by using the fast Fourier transforms, the one-dimensional power density estimates are obtained from the following equation.

i.e.

$$P_r = \Delta x \sum_{t=0}^{m} A_t * W_t \; e^{\frac{2\pi irt}{m}} \qquad \text{......(A-3)}$$

Where

Δx = sampling interval

r = $0, \pm 1, \pm 2, \ldots, \pm m$ lags

m = maximum lag or displacement

and

$$W_t = \frac{\sin \frac{\pi t}{m}}{\frac{\pi t}{m}} \qquad \text{for } t = 0, \pm 1, \ldots., \pm m \qquad \ldots\ldots(A\text{-}4)$$

The expression in Equation (A-4) represents the Daniel window which is applied in the lag domain to obtain the power density estimates. For a detailed discussion of Daniel window one may refer to GAG Report No. 9, 1955, Kanasewich (1966).

The block diagram in figure (A-1) indicates the procedure to obtain the power density estimates by using the fast Fourier transforms.

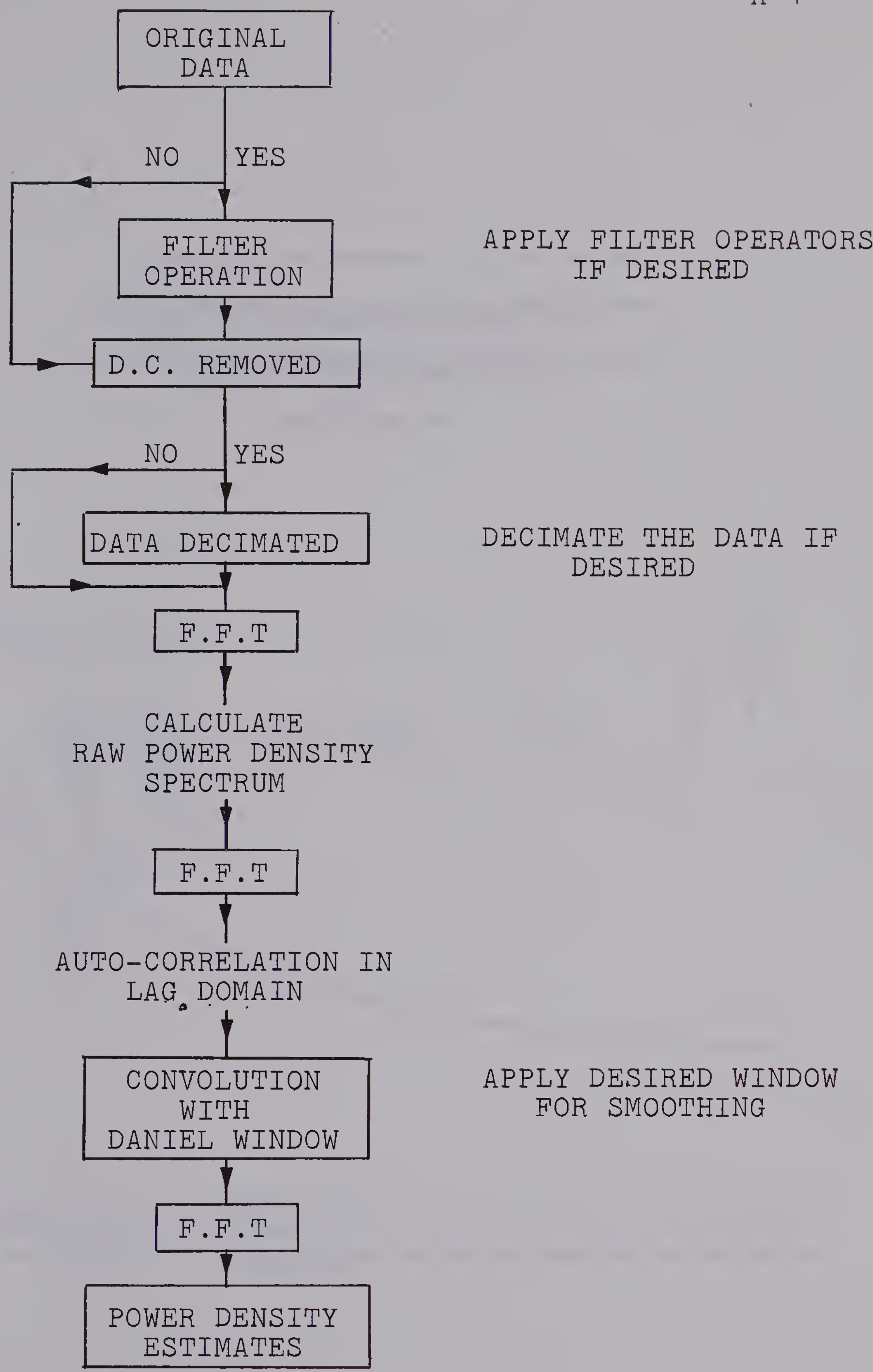


Figure A-1. Block Diagram to obtain power density estimates using fast Fourier transforms.

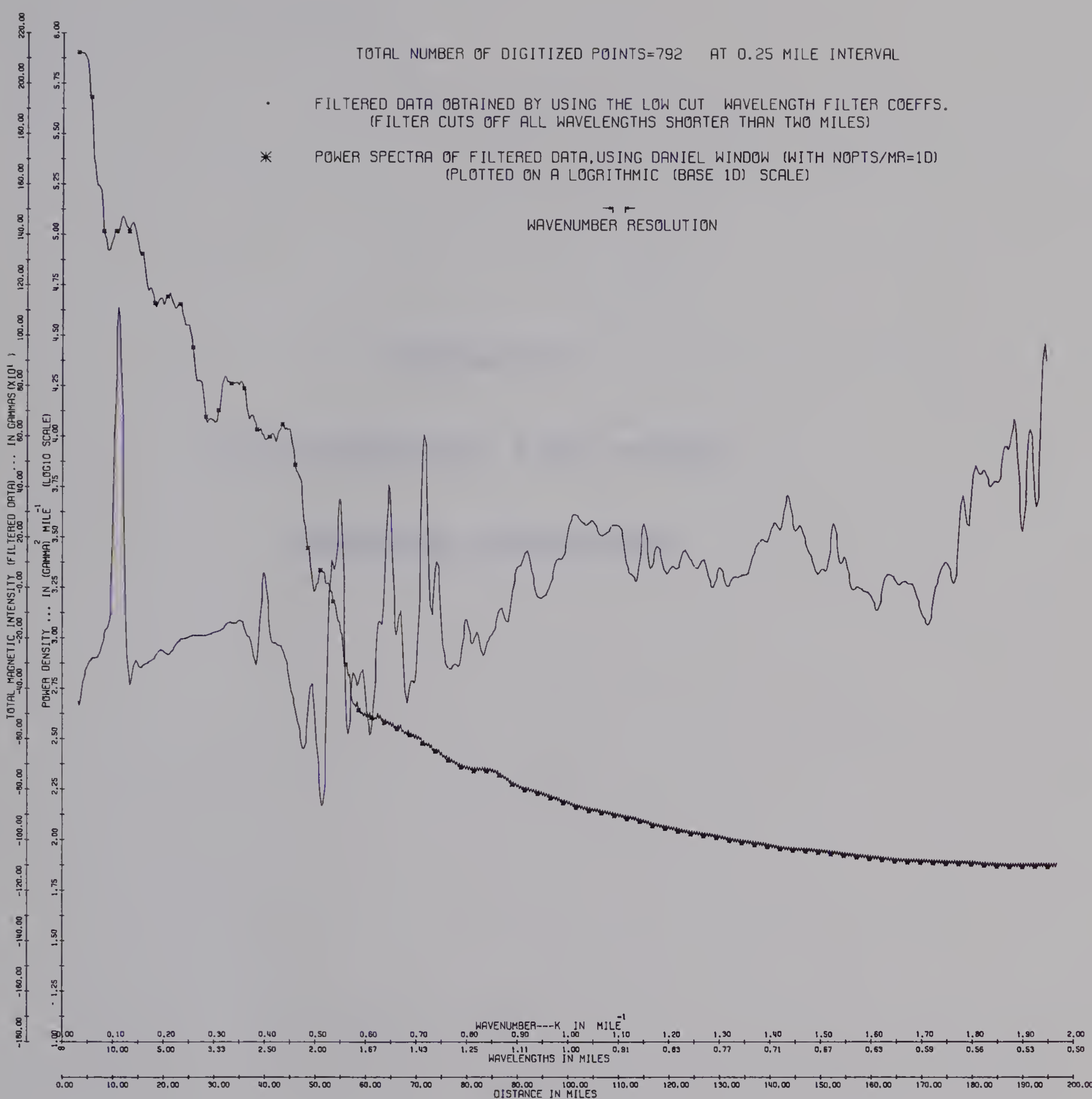


Figure A-2

Spectral density plot of a long magnetic profile (using filtered data with wavelengths shorter than 2 miles removed) at ¼ mile digital interval

FORTRAN IV

ONE DIMENSIONAL FAST FOURIER

TRANSFORM SUBROUTINES

```
      SUBROUTINE AR 1D FT (N,X,Y,S)
C     ARBITRARY RADIX ONE DIMENSIONAL FOURIER TRANSFORM
C
      INTEGER N
      REAL X (10), Y (10), S (10)
C
      CALL GR 1D FT (N,X,Y)
      CALL GR 1D FS (N,X,Y,S)
      RETURN
      END
```

```
      SUBROUTINE  GR 1D FT (NOPTS,X,Y)
C     ONE DIMENSIONAL FOURIER TRANSFORM
C
      REAL  X(10),Y(10)
C
      INTEGER J,K,M,MR,J1,J2,J3,J4,J5,JT
      REAL I1,I2,I3,I4,I5
      INTEGER P,PMAX,U,V
C
C     NEEDS SORT 1D  TO RECOVER UNSCRAMBLED FOURIER COEFFICIENTS
C
C     THIS SUBROUTINE REPLACES X + I Y BY ITS FOURIER TRANSFORN WHERE
C     X(F)+IY(F)  =  SUM T=0,(NOPTS-1) OF (X(T)+IY(T))*EXP(-F*T/NOPTS).
C
C     REAL I (PMAX), R (PMAX), C (PMAX,PMAX), S (PMAX,PMAX),
C    .A ((PMAX-1)**2+1), B ((PMAX-1)**2+1)
C
      REAL I (13), R (13), C (13,13), S (13,13), A (145), B (145)
C
      PMAX=13
C
      TWOPI=6.283185307
      M=NOPTS
  100 CONTINUE
      IF (M.NE.(M/4)*4) GO TO 400
C
C     FACTORS OF FOUR
C
      MR=M
      M=M/4
      DO 300 J=1,M
      ARG=TWOPI*FLOAT(J-1)/FLOAT(MR)
      C1=COS(ARG)
      S1=SIN(ARG)
      C2=COS(2.0*ARG)
      S2=SIN(2.0*ARG)
      C3=COS(3.0*ARG)
      S3=SIN(3.0*ARG)
      DO 200 K=MR,NOPTS,MR
      J1=J+K-MR
      J2=J1+M
      J3=J2+M
      J4=J3+M
      R1=X(J1)+X(J3)
      R2=X(J1)-X(J3)
      I1=Y(J1)+Y(J3)
      I2=Y(J1)-Y(J3)
      R3=X(J2)+X(J4)
      R4=X(J2)-X(J4)
      I3=Y(J2)+Y(J4)
      I4=Y(J2)-Y(J4)
      X(J1)=R1+R3
      Y(J1)=I1+I3
```

```
      X(J2)=(R2+I4)*C1+(I2-R4)*S1
      Y(J2)=(I2-R4)*C1-(R2+I4)*S1
      X(J3)=(R1-R3)*C2+(I1-I3)*S2
      Y(J3)=(I1-I3)*C2-(R1-R3)*S2
      X(J4)=(R2-I4)*C3+(I2+R4)*S3
      Y(J4)=(I2+R4)*C3-(R2-I4)*S3
  200 CONTINUE
  300 CONTINUE
      GO TO 100
  400 CONTINUE
      IF (M.NE.(M/2)*2) GO TO 700
C
C     FACTORS OF TWO
C
      MR=M
      M=M/2
      DO 600 J=1,M
      ARG=TWOPI*FLOAT(J-1)/FLOAT(MR)
      C1=COS(ARG)
      S1=SIN(ARG)
      DO 500 K=MR,NCPTS,MR
      J1=J+K-MR
      J2=J1+M
      R1=X(J1)+X(J2)
      R2=X(J1)-X(J2)
      I1=Y(J1)+Y(J2)
      I2=Y(J1)-Y(J2)
      X(J1)=R1
      Y(J1)=I1
      X(J2)=R2*C1+I2*S1
      Y(J2)=I2*C1-R2*S1
  500 CONTINUE
  600 CONTINUE
      GO TO 400
  700 CONTINUE
      IF (M.NE.(M/3)*3) GO TO 1000
C
C     FACTORS OF THREE
C
      MR=M
      M=M/3
      A1=COS(TWOPI/3.0)
      B1=SIN(TWOPI/3.0)
      A2=COS(2.0*TWOPI/3.0)
      B2=SIN(2.0*TWOPI/3.0)
      DO 900 J=1,M
      ARG=TWOPI*FLOAT(J-1)/FLOAT(MR)
C
C     ABSORB TWIDDLE FACTOR INTO ANALYSIS COEFFICIENTS
C
      C21=COS(ARG)
      S21=SIN(ARG)
      C22=C21*A1-S21*B1
```

```
      S22=C21*B1+S21*A1
      C23=C21*A2-S21*B2
      S23=C21*B2+S21*A2
      C31=COS(2.0*ARG)
      S31=SIN(2.0*ARG)
      C32=C31*A2-S31*B2
      S32=C31*B2+S31*A2
      C33=C31*A1-S31*B1
      S33=C31*B1+S31*A1
      DO 800 K=MR,NCPTS,MR
      J1=J+K-MR
      J2=J1+M
      J3=J2+M
      R1=X(J1)
      I1=Y(J1)
      R2=X(J2)
      I2=Y(J2)
      R3=X(J3)
      I3=Y(J3)
      X(J1)=R1+R2+R3
      Y(J1)=I1+I2+I3
      X(J2)=R1*C21+I1*S21+R2*C22+I2*S22+R3*C23+I3*S23
      Y(J2)=I1*C21-R1*S21+I2*C22-R2*S22+I3*C23-R3*S23
      X(J3)=R1*C31+I1*S31+R2*C32+I2*S32+R3*C33+I3*S33
      Y(J3)=I1*C31-R1*S31+I2*C32-R2*S32+I3*C33-R3*S33
  800 CONTINUE
  900 CONTINUE
      GO TO 700
 1000 CONTINUE
      IF (M.NE.(M/5)*5) GO TO 1300
C
C      FACTORS OF FIVE
C
      MR=M
      M=M/5
      A1=COS(TWOPI/5.0)
      B1=SIN(TWOPI/5.0)
      A2=COS(2.0*TWOPI/5.0)
      B2=SIN(2.0*TWOPI/5.0)
      A3=COS(3.0*TWOPI/5.0)
      B3=SIN(3.0*TWOPI/5.0)
      A4=COS(4.0*TWOPI/5.0)
      B4=SIN(4.0*TWOPI/5.0)
      DO 1200 J=1,M
      ARG=TWOPI*FLOAT(J-1)/FLOAT(MR)
C
C     ABSORB TWIDDLE FACTOR INTO ANALYSIS COEFFICIENTS
C
      C21=COS(ARG)
      S21=SIN(ARG)
      C22=C21*A1-S21*B1
      S22=C21*B1+S21*A1
      C23=C21*A2-S21*B2
```

```
      S23=C21*B2+S21*A2
      C24=C21*A3-S21*B3
      S24=C21*B3+S21*A3
      C25=C21*A4-S21*B4
      S25=C21*B4+S21*A4
      C31=COS(2.0*ARG)
      S31=SIN(2.0*ARG)
      C32=C31*A2-S31*B2
      S32=C31*B2+S31*A2
      C33=C31*A4-S31*B4
      S33=C31*B4+S31*A4
      C34=C31*A1-S31*B1
      S34=C31*B1+S31*A1
      C35=C31*A3-S31*B3
      S35=C31*B3+S31*A3
      C41=COS(3.0*ARG)
      S41=SIN(3.0*ARG)
      C42=C41*A3-S41*B3
      S42=C41*B3+S41*A3
      C43=C41*A1-S41*B1
      S43=C41*B1+S41*A1
      C44=C41*A4-S41*B4
      S44=C41*B4+S41*A4
      C45=C41*A2-S41*B2
      S45=C41*B2+S41*A2
      C51=COS(4.0*ARG)
      S51=SIN(4.0*ARG)
      C52=C51*A4-S51*B4
      S52=C51*B4+S51*A4
      C53=C51*A3-S51*B3
      S53=C51*B3+S51*A3
      C54=C51*A2-S51*B2
      S54=C51*B2+S51*A2
      C55=C51*A1-S51*B1
      S55=C51*B1+S51*A1
      DO 1100 K=MR,NCPTS,MR
      J1=J+K-MR
      J2=J1+M
      J3=J2+M
      J4=J3+M
      J5=J4+M
      R1=X(J1)
      I1=Y(J1)
      R2=X(J2)
      I2=Y(J2)
      R3=X(J3)
      I3=Y(J3)
      R4=X(J4)
      I4=Y(J4)
      R5=X(J5)
      I5=Y(J5)
      X(J1)=R1+R2+R3+R4+R5
      Y(J1)=I1+I2+I3+I4+I5
```

```
      X(J2)=R1*C21+I1*S21+R2*C22+I2*S22+R3*C23+I3*S23+R4*C24+I4*S24+
     .R5*C25+I5*S25
      Y(J2)=I1*C21-R1*S21+I2*C22-R2*S22+I3*C23-R3*S23+I4*C24-R4*S24+
     .I5*C25-R5*S25
      X(J3)=R1*C31+I1*S31+R2*C32+I2*S32+R3*C33+I3*S33+R4*C34+I4*S34+
     .R5*C35+I5*S35
      Y(J3)=I1*C31-R1*S31+I2*C32-R2*S32+I3*C33-R3*S33+I4*C34-R4*S34+
     .I5*C35-R5*S35
      X(J4)=R1*C41+I1*S41+R2*C42+I2*S42+R3*C43+I3*S43+R4*C44+I4*S44+
     .R5*C45+I5*S45
      Y(J4)=I1*C41-R1*S41+I2*C42-R2*S42+I3*C43-R3*S43+I4*C44-R4*S44+
     .I5*C45-R5*S45
      X(J5)=R1*C51+I1*S51+R2*C52+I2*S52+R3*C53+I3*S53+R4*C54+I4*S54+
     .R5*C55+I5*S55
      Y(J5)=I1*C51-R1*S51+I2*C52-R2*S52+I3*C53-R3*S53+I4*C54-R4*S54+
     .I5*C55-R5*S55
 1100 CONTINUE
 1200 CONTINUE
      GO TO 1000
 1300 CONTINUE
      IF (M.LE.1) GO TO 2400
C
C     GENERAL FACTORS
C
      DO 1400 J=2,PMAX
      P=J
      IF (M.EQ.(M/P)*P) GO TO 1500
 1400 CONTINUE
      CALL FCT ERR
 1500 CONTINUE
      JT=(P-1)**2+1
C
C     SET UP ARBITRARY FACTORS
C
      DO 1600 J=1,JT
      ARG=TWOPI*FLOAT(J-1)/FLOAT(P)
      A(J)=COS(ARG)
      B(J)=SIN(ARG)
 1600 CONTINUE
      MR=M
      M=M/P
      DO 2300 J=1,M
      ARG=TWOPI*FLOAT(J-1)/FLOAT(MR)
C
C     ABSORB TWIDDLE FACTOR INTO ANALYSIS COEFFICIENTS
C
      DO 1800 U=1,P
      C(U,1)=COS(FLOAT(U-1)*ARG)
      S(U,1)=SIN(FLOAT(U-1)*ARG)
      DO 1700 V=2,P
      JT=(U-1)*(V-1)+1
      C(U,V)=C(U,1)*A(JT)-S(U,1)*B(JT)
      S(U,V)=C(U,1)*B(JT)+S(U,1)*A(JT)
```

```
 1700 CONTINUE
 1800 CONTINUE
      DO 2200 K=MR,NOPTS,MR
C
C     GENERAL ANALYSIS
C
      DO 1900 U=1,P
      JT=J+K-MR+(U-1)*M
      R(U)=X(JT)
      I(U)=Y(JT)
 1900 CONTINUE
      DO 2100 U=1,P
      XT=0.0
      YT=0.0
      DO 2000 V=1,P
      XT=XT+R(V)*C(U,V)+I(V)*S(U,V)
      YT=YT+I(V)*C(U,V)-R(V)*S(U,V)
 2000 CONTINUE
      JT=J+K-MR+(U-1)*M
      X(JT)=XT
      Y(JT)=YT
 2100 CONTINUE
 2200 CONTINUE
 2300 CONTINUE
      GO TO 1300
 2400 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE GR 1D FS (NOPTS,X,Y,S)
C     UNSCRAMBLING PROGRAM FOR ONE DIMENSIONAL FOURIER COEFFICIENTS
C
      REAL  X(10),Y(10),S(10)
C
      INTEGER JT
      INTEGER DO,LIM(13),STEP(13),P,PMAX
      INTEGER A,B,C,D,E,F,G,H,I,J,K,L,M,AL,BL,CL,DL,EL,FL,GL,HL,IL,JL,
     .KL,LL,ML,AS,BS,CS,DS,ES,FS,GS,HS,IS,JS,KS,LS,MS
C
C     DIGIT REVERSER FOR USE WITH FOUR 1D . S MUST BE THE SAME SIZE AS
C     X AND Y.
C
C     EQUIVALENCES TO ALLOW INDEXING TO SET PARAMETERS AND ALLOW
C     SCALARS FOR USE IN THE DO LOOPS.
C
      EQUIVALENCE (AS,STEP(1)),(BS,STEP(2)),(CS,STEP(3)),(DS,STEP(4)),
     .(ES,STEP(5)),(FS,STEP(6)),(GS,STEP(7)),(HS,STEP(8)),(IS,STEP(9)),
     .(JS,STEP(10)),(KS,STEP(11)),(LS,STEP(12)),(MS,STEP(13))
      EQUIVALENCE (AL,LIM(1)),(BL,LIM(2)),(CL,LIM(3)),(DL,LIM(4)),
     .(EL,LIM(5)),(FL,LIM(6)),(GL,LIM(7)),(HL,LIM(8)),(IL,LIM(9)),
     .(JL,LIM(10)),(KL,LIM(11)),(LL,LIM(12)),(ML,LIM(13))
C
C     PMAX IS SET TO AGREE WITH FOUR 1D
C
      PMAX=13
C
C     SET LIMITS AND STEP SIZES FROM INNER LOOPS GOING OUT
C
      DO=13
      M=NOPTS
  100 CONTINUE
C
C     CHECK FOR FACTORS OF 4
C
      IF (M.NE.(M/4)*4) GO TO 200
      M=M/4
C
C     REALLY WANT 0-4*M-1 BUT WE GO FROM 1 TO 4*M.  4 STEPS OF M WITH
C     MAXIMUM DISPLACEMENT OF M INITIALLY
C
      LIM(DO)=4*M
      STEP(DO)=M
      DO=DO-1
      GO TO 100
  200 CONTINUE
C
C     CHECK FOR REMAINING FACTORS
C
      IF (M.LE.1) GO TO 500
C
C     FACTORS OF 2,3,5,7,11,13
C
```

```
      DO 300 JT=2,PMAX
      P=JT
      IF (M.EQ.(M/P)*P) GO TO 400
  300 CONTINUE
C
C     ERROR EXIT IF FACTORS ABOVE PMAX ARE NEEDED
C
      CALL FCT ERR
  400 CONTINUE
      M=M/P
C
C     REALLY WANT 0-P*M-1 BUT WE USE 1 TO P*M.  P STEPS OF M WITH
C     MAXIMUM INITIAL DISPLACEMENT OF M
C
      LIM(DO)=P*M
      STEP(DO)=M
      DO=DO-1
      GO TO 200
  500 CONTINUE
C
C     FINISH OUT THE DO LOOPS TO MAKE OUTER LOOPS EXECUTE ONLY ONCE
C
      DO 600 JT=1,DO
      LIM(JT)=1
      STEP(JT)=1
  600 CONTINUE
C
C     SET JT SO THAT JT RUNS FROM 1 TO NOPTS  IN STEPS OF 1 WHILE M WILL
C     RUN WITH REVERSED DIGITS
C
      JT=0
      DO 700 A=1,AL,AS
      DO 700 B=A,BL,BS
      DO 700 C=B,CL,CS
      DO 700 D=C,DL,DS
      DO 700 E=D,EL,ES
      DO 700 F=E,FL,FS
      DO 700 G=F,GL,GS
      DO 700 H=G,HL,HS
      DO 700 I=H,IL,IS
      DO 700 J=I,JL,JS
      DO 700 K=J,KL,KS
      DO 700 L=K,LL,LS
      DO 700 M=L,ML,MS
      JT=JT+1
      S(JT)=X(M)
  700 CONTINUE
C
C     COPY BACK OUT OF THE SCRATCH ARRAY
C
      DO 800 JT=1,NOPTS
      X(JT)=S(JT)
  800 CONTINUE
```

```
C
      JT=0
      DO 900 A=1,AL,AS
      DO 900 B=A,BL,BS
      DO 900 C=B,CL,CS
      DO 900 D=C,DL,DS
      DO 900 E=D,EL,ES
      DO 900 F=E,FL,FS
      DO 900 G=F,GL,GS
      DO 900 H=G,HL,HS
      DO 900 I=H,IL,IS
      DO 900 J=I,JL,JS
      DO 900 K=J,KL,KS
      DO 900 L=K,LL,LS
      DO 900 M=L,ML,MS
      JT=JT+1
      S(JT)=Y(M)
  900 CONTINUE
C
C     COPY BACK OUT OF THE SCRATCH ARRAY
C
      DO 950 JT=1,NCPTS
      Y(JT)=S(JT)
  950 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE FCT ERR
C     FACTCRING ERRCR
C
C     FACTCRING ERROR IN FCUR 1D OR SORT 1D.
C
C     CURRENTLY TAKEN IF A FACTOR ABOVE 13 IS REQUIRED.  (THE ARRAYS
C     ARE NOT BIG ENOUGH TO HANDLE THINGS ABOVE 13.)
C
      WRITE (6,100)
      CALL EXIT
  100 FORMAT (1X,15HFACTCRING ERROR)
      RETURN
      END
```

FORTRAN IV

MULTI-DIMENSIONAL FAST FOURIER

TRANSFORM SUBROUTINES

```
      SUBROUTINE AR MD FT (N,X,Y,S)
C     ARBITRARY RADIX MULTI DIMENSIONAL FOURIER TRANSFORM
C
C     FIRST SUBSCRIPT VARIES FASTEST IN KEEPING WITH FORTRAN CONVENTIONS
C
      INTEGER N (10)
      REAL X (10), Y (10), S (10)
C
      CALL GR MD FT (N,X,Y)
      CALL GR MD FS (N,X,S)
      CALL GR MD FS (N,Y,S)
      RETURN
      END
```

```
      SUBROUTINE GR MD FT (N,X,Y)
C     GENERAL RADIX MULTI DIMENSIONAL FOURIER TRANSFORM
C
      INTEGER N (10)
      REAL X (10), Y (10)
C
      INTEGER DIMEN,J,JJ,J0,J1,J2,J3,J4,K,M,MR,P,PMAX,PROD,SC,U,V
      REAL ARG,A1,A2,A3,A4,B1,B2,B3,B4,C1,C2,C3,C10,C11,C12,C13,C14,
     .C20,C21,C22,C23,C24,C30,C31,C32,C33,C34,C40,C41,C42,C43,C44,I0,I1,
     .I2,I3,I4,R0,R1,R2,R3,R4,S1,S2,S3,S10,S11,S12,S13,S14,S20,S21,
     .S22,S23,S24,S30,S31,S32,S33,S34,S40,S41,S42,S43,S44,TWOPI,XT,YT
C
      REAL A (19), B (19), C (19,19), I (19), R (19), S (19,19)
C
      PMAX=19
C
      TWOPI=6.283185307
      DIMEN=1
      PROD=1
  100 CONTINUE
      PROD=PROD*N(DIMEN)
      DIMEN=DIMEN+1
      IF (N(DIMEN).GT.0) GO TO 100
      DIMEN=DIMEN-1
      SC=PROD
      M=PROD
  200 CONTINUE
      SC=SC/N(DIMEN)
      DIMEN=DIMEN-1
  300 CONTINUE
      IF (M/SC.NE.M/SC/4*4) GO TO 600
      MR=M
      M=M/4
      DO 500 J=1,M
      ARG=TWOPI*FLOAT((J-1)/SC)/FLOAT(MR/SC)
      C1=COS(ARG)
      S1=SIN(ARG)
      C2=COS(2.0*ARG)
      S2=SIN(2.0*ARG)
      C3=COS(3.0*ARG)
      S3=SIN(3.0*ARG)
      DO 400 K=MR,PROD,MR
      J0=J+K-MR
      J1=J0+M
      J2=J1+M
      J3=J2+M
      R0=X(J0)+X(J2)
      R1=X(J0)-X(J2)
      I0=Y(J0)+Y(J2)
      I1=Y(J0)-Y(J2)
      R2=X(J1)+X(J3)
      R3=X(J1)-X(J3)
      I2=Y(J1)+Y(J3)
```

```
      I3=Y(J1)-Y(J3)
      X(J0)=R0+R2
      Y(J0)=I0+I2
      X(J2)=(R1+I3)*C1+(I1-R3)*S1
      Y(J2)=(I1-R3)*C1-(R1+I3)*S1
      X(J1)=(R0-R2)*C2+(I0-I2)*S2
      Y(J1)=(I0-I2)*C2-(R0-R2)*S2
      X(J3)=(R1-I3)*C3+(I1+R3)*S3
      Y(J3)=(I1+R3)*C3-(R1-I3)*S3
  400 CONTINUE
  500 CONTINUE
      GO TO 300
  600 CONTINUE
      IF (M/SC.NE.M/SC/2*2) GO TO 900
      MR=M
      M=M/2
      DO 800 J=1,M
      ARG=TWOPI*FLOAT((J-1)/SC)/FLOAT(MR/SC)
      C1=COS(ARG)
      S1=SIN(ARG)
      DO 700 K=MR,PROD,MR
      J0=J+K-MR
      J1=J0+M
      R0=X(J0)+X(J1)
      R1=X(J0)-X(J1)
      I0=Y(J0)+Y(J1)
      I1=Y(J0)-Y(J1)
      X(J0)=R0
      Y(J0)=I0
      X(J1)=R1*C1+I1*S1
      Y(J1)=I1*C1-R1*S1
  700 CONTINUE
  800 CONTINUE
      GO TO 600
  900 CONTINUE
      IF (M/SC.NE.M/SC/3*3) GO TO 1200
      MR=M
      M=M/3
      A1=COS(TWOPI/3.0)
      B1=SIN(TWOPI/3.0)
      A2=COS(2.0*TWOPI/3.0)
      B2=SIN(2.0*TWOPI/3.0)
      DO 1100 J=1,M
      ARG=TWOPI*FLOAT((J-1)/SC)/FLOAT(MR/SC)
      C10=COS(ARG)
      S10=SIN(ARG)
      C11=C10*A1-S10*B1
      S11=C10*B1+S10*A1
      C12=C10*A2-S10*B2
      S12=C10*B2+S10*A2
      C20=COS(2.0*ARG)
      S20=SIN(2.0*ARG)
      C21=C20*A2-S20*B2
```

```
      S21=C20*B2+S20*A2
      C22=C20*A1-S20*B1
      S22=C20*B1+S20*A1
      DO 1000 K=MR,PROD,MR
      J0=J+K-MR
      J1=J0+M
      J2=J1+M
      R0=X(J0)
      I0=Y(J0)
      R1=X(J1)
      I1=Y(J1)
      R2=X(J2)
      I2=Y(J2)
      X(J0)=R0+R1+R2
      Y(J0)=I0+I1+I2
      X(J1)=R0*C10+I0*S10+R1*C11+I1*S11+R2*C12+I2*S12
      Y(J1)=I0*C10-R0*S10+I1*C11-R1*S11+I2*C12-R2*S12
      X(J2)=R0*C20+I0*S20+R1*C21+I1*S21+R2*C22+I2*S22
      Y(J2)=I0*C20-R0*S20+I1*C21-R1*S21+I2*C22-R2*S22
 1000 CONTINUE
 1100 CONTINUE
      GO TO 900
 1200 CONTINUE
      IF (M/SC.NE.M/SC/5*5) GO TO 1500
      MR=M
      M=M/5
      A1=COS(TWOPI/5.0)
      B1=SIN(TWOPI/5.0)
      A2=COS(2.0*TWOPI/5.0)
      B2=SIN(2.0*TWOPI/5.0)
      A3=COS(3.0*TWOPI/5.0)
      B3=SIN(3.0*TWOPI/5.0)
      A4=COS(4.0*TWOPI/5.0)
      B4=SIN(4.0*TWOPI/5.0)
      DO 1400 J=1,M
      ARG=TWOPI*FLOAT((J-1)/SC)/FLOAT(MR/SC)
      C10=COS(ARG)
      S10=SIN(ARG)
      C11=C10*A1-S10*B1
      S11=C10*B1+S10*A1
      C12=C10*A2-S10*B2
      S12=C10*B2+S10*A2
      C13=C10*A3-S10*B3
      S13=C10*B3+S10*A3
      C14=C10*A4-S10*B4
      S14=C10*B4+S10*A4
      C20=COS(2.0*ARG)
      S20=SIN(2.0*ARG)
      C21=C20*A2-S20*B2
      S21=C20*B2+S20*A2
      C22=C20*A4-S20*B4
      S22=C20*B4+S20*A4
      C23=C20*A1-S20*B1
```

```
      S23=C20*B1+S20*A1
      C24=C20*A3-S20*B3
      S24=C20*B3+S20*A3
      C30=COS(3.0*ARG)
      S30=SIN(3.0*ARG)
      C31=C30*A3-S30*B3
      S31=C30*B3+S30*A3
      C32=C30*A1-S30*B1
      S32=C30*B1+S30*A1
      C33=C30*A4-S30*B4
      S33=C30*B4+S30*A4
      C34=C30*A2-S30*B2
      S34=C30*B2+S30*A2
      C40=COS(4.0*ARG)
      S40=SIN(4.0*ARG)
      C41=C40*A4-S40*B4
      S41=C40*B4+S40*A4
      C42=C40*A3-S40*B3
      S42=C40*B3+S40*A3
      C43=C40*A2-S40*B2
      S43=C40*B2+S40*A2
      C44=C40*A1-S40*B1
      S44=C40*B1+S40*A1
      DO 1300 K=MR,PROD,MR
      J0=J+K-MR
      J1=J0+M
      J2=J1+M
      J3=J2+M
      J4=J3+M
      R0=X(J0)
      I0=Y(J0)
      R1=X(J1)
      I1=Y(J1)
      R2=X(J2)
      I2=Y(J2)
      R3=X(J3)
      I3=Y(J3)
      R4=X(J4)
      I4=Y(J4)
      X(J0)=R0+R1+R2+R3+R4
      Y(J0)=I0+I1+I2+I3+I4
      X(J1)=R0*C10+I0*S10+R1*C11+I1*S11+R2*C12+I2*S12+R3*C13+I3*S13+
     .R4*C14+I4*S14
      Y(J1)=I0*C10-R0*S10+I1*C11-R1*S11+I2*C12-R2*S12+I3*C13-R3*S13+
     .I4*C14-R4*S14
      X(J2)=R0*C20+I0*S20+R1*C21+I1*S21+R2*C22+I2*S22+R3*C23+I3*S23+
     .R4*C24+I4*S24
      Y(J2)=I0*C20-R0*S20+I1*C21-R1*S21+I2*C22-R2*S22+I3*C23-R3*S23+
     .I4*C24-R4*S24
      X(J3)=R0*C30+I0*S30+R1*C31+I1*S31+R2*C32+I2*S32+R3*C33+I3*S33+
     .R4*C34+I4*S34
      Y(J3)=I0*C30-R0*S30+I1*C31-R1*S31+I2*C32-R2*S32+I3*C33-R3*S33+
     .I4*C34-R4*S34
```

```
      X(J4)=R0*C40+I0*S40+R1*C41+I1*S41+R2*C42+I2*S42+R3*C43+I3*S43+
     .R4*C44+I4*S44
      Y(J4)=I0*C40-R0*S40+I1*C41-R1*S41+I2*C42-R2*S42+I3*C43-R3*S43+
     .I4*C44-R4*S44
 1300 CONTINUE
 1400 CONTINUE
      GO TO 1200
 1500 CONTINUE
      IF (M.LE.SC) GO TO 2600
      DO 1600 J=2,PMAX
      P=J
      IF (M/SC.EQ.M/SC/P*P) GO TO 1700
 1600 CONTINUE
      GO TO 2800
 1700 CONTINUE
      MR=M
      M=M/P
      DO 1800 U=1,P
      ARG=TWOPI*FLOAT(U-1)/FLOAT(P)
      A(U)=COS(ARG)
      B(U)=SIN(ARG)
 1800 CONTINUE
      DO 2500 J=1,M
      ARG=TWOPI*FLOAT((J-1)/SC)/FLOAT(MR/SC)
      DO 2000 U=1,P
      C(U,1)=COS(FLOAT(U-1)*ARG)
      S(U,1)=SIN(FLOAT(U-1)*ARG)
      DO 1900 V=2,P
      JJ=(U-1)*(V-1)-(U-1)*(V-1)/P*P+1
      C(U,V)=C(U,1)*A(JJ)-S(U,1)*B(JJ)
      S(U,V)=C(U,1)*B(JJ)+S(U,1)*A(JJ)
 1900 CONTINUE
 2000 CONTINUE
      DO 2400 K=MR,PROD,MR
      DO 2100 U=1,P
      JJ=J+K-MR+(U-1)*M
      R(U)=X(JJ)
      I(U)=Y(JJ)
 2100 CONTINUE
      DO 2300 U=1,P
      XT=0.0
      YT=0.0
      DO 2200 V=1,P
      XT=XT+R(V)*C(U,V)+I(V)*S(U,V)
      YT=YT+I(V)*C(U,V)-R(V)*S(U,V)
 2200 CONTINUE
      JJ=J+K-MR+(U-1)*M
      X(JJ)=XT
      Y(JJ)=YT
 2300 CONTINUE
 2400 CONTINUE
 2500 CONTINUE
      GO TO 1500
```

```
2600 CONTINUE
     IF (DIMEN.GT.0) GO TO 200
2700 CONTINUE
     RETURN
2800 CONTINUE
     DIMEN=DIMEN+1
     WRITE (6,2900) DIMEN,N(DIMEN)
     GO TO 2700
2900 FORMAT (1X,28HFACTORING ERROR IN DIMENSION,I2,10H.  N(DIM)=,I5)
     END
```

```
      SUBROUTINE GR MD FS (PTS,X,T)
C     GENERAL RADIX MULTI DIMENSIONAL FOUTIER SORT
C
      INTEGER PTS (10)
      REAL X (10), T (10)
C
      INTEGER DIMEN,DO,II,JJ,P,PMAX,SC, S (19), U (19)
      INTEGER A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,Q,R,V,W,AL,BL,CL,DL,EL,FL,GL
     .,HL,IL,JL,KL,LL,ML,NL,OL,QL,RL,VL,WL,AS,BS,CS,DS,ES,FS,GS,HS,IS,JS
     .,KS,LS,MS,NS,OS,QS,RS,VS,WS
      EQUIVALENCE (AS,S(1)),(BS,S(2)),(CS,S(3)),(DS,S(4)),(ES,S(5)),
     .(FS,S(6)),(GS,S(7)),(HS,S(8)),(IS,S(9)),(JS,S(10)),(KS,S(11)),
     .(LS,S(12)),(MS,S(13)),(NS,S(14)),(OS,S(15)),(QS,S(16)),(RS,S(17)),
     .(VS,S(18)),(WS,S(19)),(AL,U(1)),(BL,U(2)),(CL,U(3)),(DL,U(4)),(EL,
     .U(5)),(FL,U(6)),(GL,U(7)),(HL,U(8)),(IL,U(9)),(JL,U(10)),(KL,U(11)
     .),(LL,U(12)),(ML,U(13)),(NL,U(14)),(OL,U(15)),(QL,U(16)),(RL,U(17)
     .),(VL,U(18)),(WL,U(19))
C
      PMAX=19
C
      DO=19
      DIMEN=1
      SC=1
  100 CONTINUE
      M=PTS(DIMEN)
  200 CONTINUE
      IF (M.LE.1) GO TO 500
      DO 300 J=2,PMAX
      P=J
      IF (M.EQ.M/P*P) GO TO 400
  300 CONTINUE
      GO TO 1100
  400 CONTINUE
      U(DO)=M*SC
      S(DO)=M/P*SC
      M=M/P
      DO=DO-1
      GO TO 200
  500 CONTINUE
      SC=SC*PTS(DIMEN)
      DIMEN=DIMEN+1
      IF (PTS(DIMEN).GT.0) GO TO 100
      IF (DO.LE.0) GO TO 700
      DO 600 J=1,DO
      U(J)=1
      S(J)=1
  600 CONTINUE
  700 CONTINUE
      JJ=0
      DO 800 A=1,AL,AS
      DO 800 B=1,BL,BS
      DO 800 C=1,CL,CS
      DO 800 D=1,DL,DS
```

```
      DO 800 E=1,EL,ES
      DO 800 F=1,FL,FS
      DO 800 G=1,GL,GS
      DO 800 H=1,HL,HS
      DO 800 I=1,IL,IS
      DO 800 J=1,JL,JS
      DO 800 K=1,KL,KS
      DO 800 L=1,LL,LS
      DO 800 M=1,ML,MS
      DO 800 N=1,NL,NS
      DO 800 O=1,OL,OS
      DO 800 Q=1,QL,QS
      DO 800 R=1,RL,RS
      DO 800 V=1,VL,VS
      DO 800 W=1,WL,WS
      II=A+B+C+D+E+F+G+H+I+J+K+L+M+N+O+Q+R+V+W-18
      JJ=JJ+1
      T(JJ)=X(II)
  800 CONTINUE
      DO 900 J=1,SC
      X(J)=T(J)
  900 CONTINUE
 1000 CONTINUE
      RETURN
 1100 CONTINUE
      WRITE (6,1200) DIMEN,PTS(DIMEN)
      GO TO 1000
 1200 FORMAT (1X,28HFACTORING ERROR IN DIMENSION,I2,10H.  N(DIM)=,I5)
      END
```

FORTRAN IV

ONE DIMENSIONAL AUTO-CORRELATION AND AUTO-POWER SPECTRA CALCULATIONS USING DANIEL WINDOW AND FAST FOURIER TRANSFORM SUBROUTINES. THE PROGRAM FILTERS THE DATA ALSO, IF REQUIRED.

NPT		Total number of data points digitized at an interval of ¼ units.
NOPTS		Number of points after re-digitization, if required, at ½ or ¾ units apart.
MR		For Daniel window 'MR' is approximately 10% of 'NOPTS'.
FMT(1)		Defines FORMAT of Input data as (1X,18I4,7X).
DELT		Required digitizing interval.
IX(NPT) ...		Vector to read the input data.
ICC		Total number of operations to be carried out.
ISS	= 1	if auto-correlation and spectral analysis are required on unfiltered data.
	= 0	if required on filtered data.
IS	= 1	if using highpass wave number filter coefficients.
	=-1	if using lowpass wave number filter coefficients
	= 0	if using bandpass wave number filter coefficients.
LF		Number of filter coefficient terms.
FRE1		Low-cut wave number of the filter.

FRE2	High-cut wave number of the filter.
FRE1 = FRE2	if using lowpass or highpass filter coefficients.
F(LF)	Vector to read in the filter coefficients.

READ IN THE FOLLOWING PARAMETERS

1.	NPT,NOPTS,MR	according to FORMAT(5X,10I5)
2.	FMT(12)	according to FORMAT(1X,18A4)
3.	DELT	according to FORMAT(5X,F10.5)
4.	IX(NPT)	will be read in accordance to FMT
5.	ICC	according to FORMAT(5X,10I5)
6.	ISS,IS	according to FORMAT(5X,10I5)
7.	LF,FRE1,FRE2	according to FORMAT(5X,I5,2F10.5)
8.	F(LF)	according to FORMAT(1X,4E16.8)

```
C      ONE DIMENSIONAL AUTO-CORRELATION AND AUTO-POWER SPECTRA USING
C      DANIEL WINDOW AND FAST FOURIER TRANSFORM SUBROUTINES.
C
C      THIS PROGRAM CALCULATES AUTO POWER DENSITY SPECTRA AND AUTO
C      CORRELATION OF INPUT DATA WITH VARIABLE FORMATS
C
C      ALSO WE CONVOLVE THE INPUT DATA WITH FILTER COEFFICIENTS AND THEN
C      IN TURN OBTAIN THE AUTO POWER DENSITY SPECTRA AND AUTO-CORRELATION
C
C      WHEN FILTER IS USED LET US SAY OF LENGTH LF, THEN THE FINAL USEFUL
C      DATA IS REDUCED BY LF/2 AT EACH END OF THE RECORD.
C
C      DATA SET IS READ IN VECTOR IX(4096) OF KNOWN POINTS NPT.
C      FILTER COEFFICIENTS ARE READ IN VECTOR F(200) OF KNOWN TERMS LF.
C      NPT     ***   IS THE TOTAL NUMBER OF DATA POINTS AT DELT=0.25 MILE
C      ICC     ***   TOTAL NUMBER OF OPERATIONS TO BE CARRIED OUT.
C      LF      ***   NUMBER OF FILTER COEFFICIENT TERMS.
C      FRE1    ***   LOW CUT FREQUENCY OF THE FILTER.
C      FRE2    ***   HIGH CUT FREQUENCY OF THE FILTER.
C      WHEN USING EITHER HIGH OR LOW PASS FREQUENCY FILTER COEFFS. WE USE
C      FRE1=FRE2
C      FOR BAND PASS FILTER WE SPECIFY FRE1 AND FRE2 AS LOW AND HIGH CUT
C      FREQUENCY VALUES OF THE FILTER.
C      IS =1   ***   MEANS USING HIGH PASS FREQUENCY FILTER COEFFS.
C      IS =-1  ***   MEANS USING LOW PASS FREQUENCY FILTER COEFFS.
C      IS =0   ***   MEANS USING BAND PASS FREQUENCY FILTER COEFFS.
C      ISS =1        AUTO-SPECTRA AND AUTO-CORRELATION ARE OBTAINED WITH-
C                    OUT FILTERING.
C      ISS =0        AUTO-SPECTRA AND AUTO-CORRELATION ARE OBTAINED AFTER
C                    FILTERING.
C      DELT    *** THE DIGITIZING INTERVAL IN SECONDS OR MILES
C      NOPTS   ***   THE NUMBER OF POINTS OF THE DATA IX(I)
C      LA,LB   ***   THE NUMBER OF POINTS AFTER CONVOLUTION.
C      FMT DEFINES FORMAT OF INPUT DATA (1X,18I4,7X)
C      FREQUENCY = J*FP.
C      FR = FREQUENCY RESOLUTION.
C
       REAL  X(4096), F(200) ,S(4096) ,Y(4096)
       INTEGER  IX(1000)
       INTEGER  N, ONE
       DIMENSION FMT(12)
       DIMENSION  XE(768)
       EQUIVALENCE   (X,XE)
C
C       DATA AND RESULT FORMATS
C
  100 FORMAT( 8(I5, E11.4) )
  115 FORMAT(1H1,57X,15HPOWER SPECTRUM /)
  130 FORMAT(1H1,50X,26HAUTO-CORRELATION FUNCTION /)
C
C       INITIAL PARAMETERS
C
       ONE = 1
```

```
      LOG2N = 10
      N = 2**LOG2N
      LOG2NP = 11
      NPRIME = 2**LOG2NP
C
      PI = 3.141592654
C
C     MR FOR DANIEL WINDOW IS CHOSEN FROM  MR=NOPTS/10
C     READ IN THE DATA
C
      READ(5,1)   NPT,NOPTS,MR
      READ(5,8)(FMT(J),J=1,12)
      READ (5,9) DELT
    1 FORMAT(5X,10I5)
    2 FORMAT(1HJ,10X,7HXMIN = ,E11.4,5X,7HXMAX = ,E11.4)
    8 FORMAT (1X,18A4)
    9 FORMAT(5X,F10.5)
      READ(5,FMT)  (IX(J),J=1,NPT)
C     THIS PROGRAM RE-DIGITIZEZES THE INPUT DATA (WHICH HAS BEEN READ IN
C     AT DELT=0.25) ACCORDING TO THE DESIRED INTERVAL OF DELT=0.5,0.75,1
      CALL  DIGIT(IX,DELT,NOPTS)
C
C
C     TO PRINT THE INPUT DATA
C
      CALL  OUTPUT(IX,NOPTS)
C
C     WRITE THE INPUT DATA FOR CALCOMP PLOTTER
C
C     READ IN THE NUMBER OF OPERATIONS TO BE CARRIED OUT
C
      READ(5,1)  ICC
      DO 200 IK=1,ICC
C
C     READ IN DATA FOR THE FILTER SPECIFICATIONS
C
      READ(5,1)  ISS,IS
C     TRANSFER CONTROL IF WE NEED AUTO-CORRELATION AND POWER SPECTRA
C     ON UNFILTERED DATA.
      IF(ISS.EQ.1)  GO TO 50
      READ(5,19)  LF,FRE1,FRE2
   19 FORMAT(5X,I5,2F10.5)
      READ(5,20)  (F(I),I=1,LF)
   20 FORMAT(1X,4E16.8)
      ALAMD1=1./FRE1
      ALAMD2=1./FRE2
      WRITE(6,21)  LF
   21 FORMAT(1H1,2X,24HNUMBER OF FILTER TERMS =,I5)
C
C     TO WRITE THE FILTER FREQ., TITLE ETC.
C
      CALL  L H B (IS,FRE1,FRE2,ALAMD1,ALAMD2)
C
```

```
C       TO PRINT THE OUTPUT OF FILTERED DATA
C       IT WRITES OUT THE FILTER COEFFS. AND THE FILTERED DATA.
C
        CALL  FILOUT(IX,NOPTS,F,LF,X,LA)
C
C       TO CUT OFF HALF THE END EFFECTS OF THE FILTERED DATA
C
        CALL  ADJUST(X,NOPTS,LF,LA)
        NN=NOPTS
C
C       SUBROUTINE DC OUT TAKES OUT THE D.C. VALUE AND WRITES OUT THE DATA
C
        CALL  DC OUT(X,NN,ISS)
C
C       MULTIPLY BY A FACTOR 10.0 BECAUSE IN PUNCHING THE CARDS FROM DIGIT
C       ORIGINAL DATA THE LAST DIGIT (I.E. OF FACTOR 10) WAS OMITED.
        DO 10 I=1,NN
        X(I)=X(I)*10.0
   10   CONTINUE
C
C       REMOVE 12 DATA POINTS FROM EACH END TO AVOID END EFFECTS
C       OF THE FILTER.
        DO 22 I=1,NN
        I1=I+12
        IF(I.GT.(NN-24))  GO TO 22
        X(I)=X(I1)
   22   CONTINUE
        NN=NN-24
C
C
C       TO WRITE THE FILTERED DATA  FOR CALCOMP PLOTTER
        WRITE(1)  XE
        XN=NN
        CALL  MAXMIN (X,XMIN,XMAX,NN)
        WRITE(6,2)  XMIN,XMAX
        GO TO 57
C       CONTROL IS TRANSFERED HERE FOR AUTO-CORRELATION AND POWER SPECTRA
C       ON UNFILTERED DATA
   50 CONTINUE
        DO 3 I=1,NOPTS
        X(I)=FLOAT(IX(I))
    3 CONTINUE
C
C       SUBROUTINE DC OUT TAKES OUT THE D.C. VALUE AND WRITES OUT THE DATA
C
        CALL  DC OUT(X,NOPTS,ISS)
        NN=NOPTS
C       MULTIPLY BY A FACTOR 10.0 BECAUSE IN PUNCHING THE CARDS FROM
C       ORIGINAL DATA THE LAST DIGIT (I.E. OF FACTOR 10) WAS OMITED.
        DO 11 I=1,NN
        X(I)=X(I)*10.0
   11   CONTINUE
C
```

```
C     WRITE THE D.C REMOVED DATA  FOR CALCOMP PLOTTER
C
      WRITE(1)  XE
      XN=NN
      CALL  MAXMIN (X,XMIN,XMAX,NN)
      WRITE(6,2)  XMIN,XMAX
   57 NN1=NN+1
      NN2=NN*2
C
      CALL  ZERO (X,NN1,NPRIME)
C
C      COMPUTE THE AUTO-CORRELATIONS CX(T)
C
      CALL  ZERO (Y,1,NPRIME)
      CALL  ZERO(S,1,NPRIME)
      CALL  AR 1D FT (NPRIME,X,Y,S)
      CALL  POWER (NPRIME,X,Y)
C
C     TO OBTAIN INVERSE FOURIER TRANSFORM.
C     X(J) IS REAL, SO WE NEED NOT TAKE COMPLEX CONJUGATE, AND
C     WE PROCEED DIRECT TO OBTAIN THE FOURIER TRANSFORMS.
C
      CALL  ZERO(S,1,NPRIME)
      CALL  AR 1D FT (NPRIME,X,Y,S)
      CALL  SCALE (N,X,FLOAT(NPRIME))
      CALL  DAN FAC (NN,X)
C
C     PRINT OUT AUTO-CORRELATIONS
C
      WRITE (6,130)
      WRITE (6,100) ((J, X(J)), J=ONE,N)
C     WRITE DATA  FOR CALCOMP PLOTTER
      CALL  MAXMIN (X,XMIN,XMAX,NN)
      WRITE(6,2)  XMIN,XMAX
C
      CALL  DANLAG(NN,MR,X)
      CALL  LAGS (NN,X)
C      COMPUTE THE POWER SPECTRUM
C
      CALL  ZERO (Y,1,NN2)
      CALL  ZERO(S,1,NN2)
      CALL  AR 1D FT (NN2,X,Y,S)
C
C      OUTPUT THE POWER SPECTRUM FUNCTION
C
C     MULTIPY POWER BY A FACTOR  DELT
      CALL  FACTOR (X,NN,DELT)
      WRITE (6,115)
      WRITE (6,100) ((J, X(J)), J=ONE,NN)
C
      DO 4 J=ONE,NN
      IF(X(J).EQ.0.0)  GO TO 4
      X(J)=ALOG10(ABS(X(J)))
```

```
   4   CONTINUE
C
C      WRITE DATA  FOR CALCOMP PLOTTER
       WRITE(1)  XE
       CALL  MAXMIN (X,XMIN,XMAX,NN)
       WRITE(6,2)  XMIN,XMAX
  200  CONTINUE
       END FILE 1
       STOP
       E  N  D
```

```
      SUBROUTINE  DIGIT(IX,DELT,NOPTS)
      INTEGER  IX(10)
C
C     THIS PROGRAM RE-DIGITIZEZES THE INPUT DATA (WHICH HAS BEEN READ IN
C     AT DELT=0.25) ACCORDING TO THE DESIRED INTERVAL OF DELT=0.5,0.75,1
C
      IF(DELT.EQ.0.25)  GO TO 40
      IF(DELT.EQ.1.0)  GO TO 45
      IF(DELT.EQ.0.75)  GO TO 46
      DO 33 J=1,NOPTS
      J1=2*J-1
   33 IX(J)=IX(J1)
      GO TO 40
   46 DO 34 J=1,NOPTS
      J1=3*J-2
   34 IX(J)=IX(J1)
      GO TO 40
   45 DO 4 J=1,NOPTS
      J1=4*J-3
    4 IX(J)=IX(J1)
   40 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE OUTPUT(IX,NOPTS)
      INTEGER  IX(10)
C     TO PRINT OUT THE DATA
      WRITE (6,17)
   17 FORMAT(1HL,59X,13HINPUT DATA IX)
      WRITE(6,18)
   18 FORMAT(1HT,59X,13H*************)
      WRITE (6,29) (IX(J),J=1,NOPTS)
   29 FORMAT(30X,18I4,7X)
      RETURN
      END
```

```
      SUBROUTINE  L H B (IS,FRE1,FRE2,ALAMD1,ALAMD2)
C
C     IT WRITES OUT LOW, HIGH, OR BAND PASS FILTER TITLES AS REQUIRED.
C
      IF(IS.EQ.1) GO TO 22
      IF(IS.EQ.0) GO TO 23
C
C    PRINT FOR LOW PASS FREQUENCY FILTER
C
      WRITE(6,60) FRE2
   60 FORMAT(1HJ,2X,29HHIGH CUT FREQUENCY (IN CPM) =,F10.5)
      WRITE(6,61) ALAMD2
   61 FORMAT(1HJ,2X,33HHERE THE WAVELENGTHS SMALLER THAN ,F5.1,1X,17HMIL
     1ES ARE CUT OFF)
      WRITE(6,62)
   62 FORMAT(1HL,35X,60HLOW PASS FREQUENCY OR LOW CUT WAVELENGTH FILTER
     1COEFFICIENTS)
      WRITE(6,63)
   63 FORMAT(1HT,35X,60H*************************************************
     1********* **)
      GO TO 41
C
C    PRINT FOR HIGH PASS FREQUENCY FILTER
C
   22 WRITE(6,70) FRE1
   70 FORMAT(1HJ,2X,28HLOW CUT FREQUENCY (IN CPM) =,F10.5)
      WRITE(6,71) ALAMD1
   71 FORMAT(1HJ,2X,33HHERE THE WAVELENGTHS GREATER THAN ,F5.1,1X,17HMIL
     1ES ARE CUT OFF)
      WRITE(6,72)
   72 FORMAT(1HL,34X,62HHIGH PASS FREQUENCY OR HIGH CUT WAVELENGTH FILTE
     1R COEFFICIENTS)
      WRITE(6,73)
   73 FORMAT(1HT,34X,62H*************************************************
     1********M****)
      GO TO 41
C
C    PRINT FOR BAND PASS FREQUENCY FILTER
C
   23 WRITE(6,70) FRE1
      WRITE(6,60) FRE2
      WRITE(6,81)  ALAMD2,ALAMD1
   81 FORMAT(1HJ,2X,33HHERE ONLY THE WAVELENGTHS BETWEEN,F5.1,1X,9HMILES
     1 AND,F5.1,1X,18HMILES ARE RETAINED)
      WRITE(6,82)
   82 FORMAT(1HL,50X,29HBAND PASS FILTER COEFFICIENTS)
      WRITE(6,83)
   83 FORMAT(1HT,50X,29H*****************************)
   41 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE  FILOUT(IX,NOPTS,F,LF,X,LA)
      INTEGER  IX(10)
      REAL  F(10),X(10)
C
C     IT WRITES OUT THE FILTER COEFFS. AND THE FILTERED DATA.
C
      WRITE(6,24)  (F(I),I=1,LF)
   24 FORMAT(1X,8E16.8)
      CALL  CONVOL(IX,NOPTS,F,LF,X,LA)
      WRITE(6,25)  LA
   25 FORMAT(1HL,51X,26HNUMBER OF FILTERED TERMS =,I5)
      WRITE(6,26)
   26 FORMAT(1X,54X,24HOUTPUT (FILTERED) DATA X)
      WRITE(6,27)
   27 FORMAT(1HT,54X,24H************************)
      WRITE(6,39)  (X(I),I=1,LA)
   39 FORMAT(12X,16F6.0,7X)
      RETURN
      END
```

```
      SUBROUTINE  CONVOL (IX,LX,F,LF,X,LA)
      REAL  X(10), F(10)
      INTEGER  IX(10)
      LA=LX+LF-1
      DO 1 I=1,LA
      X(I)=0.0
    1 CONTINUE
      DO 2 I=1,LX
      DO 2 J=1,LF
      K=I+J-1
    2 X(K)=FLOAT(IX(I))*F(J)+X(K)
      RETURN
      END
```

```
      SUBROUTINE  ADJUST (X,NOPTS,LF,LA)
      REAL  X(10)
C
C     IT CUTS OFF HALF THE END EFFECTS OF THE FILTERED DATA
C
      LF1=LF/2
      LF2=NOPTS+LF1+1
      DO 42 I=1,LF1
   42 X(I)=0.0
      DO 43 I=LF2,LA
   43 X(I)=0.0
      DO 44 I=1,NOPTS
      J=LF1+I
      X(I)=X(J)
   44 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE  ZERO (X,K,L)
      REAL  X(10)
C     THIS SETS THE VECTOR X TO ZERO
      DO 1 I=K,L
      X(I)=0.0
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE  POWER (N,X,Y)
C
C     WE CALCULATE POWER AND STORE BACK IN X
C
      REAL  X(10),Y(10)
      REAL  P
      DO 1 J=1,N
      P=X(J)**2+Y(J)**2
      X(J)=P
      Y(J)=0.0
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE  DAN FAC (N,X)
      REAL  X(10)
C
C     IT DIVIDES  THE AUTO-CORRELATION BY A FACTOR OF N IN THE CASE
C     OF DANIEL WINDOW.
C
      DO 1 I=1,N
      X(I)=X(I)/FLOAT(N)
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE  DC OUT(X,NOPTS,ISS)
      REAL  X(10)
      SX=0.
      DO 3 J=1,NOPTS
    3 SX=SX+X(J)
      SX = SX/(FLOAT(NOPTS))
      DO 4 J=1,NOPTS
    4 X(J)=X(J)-SX
      IF(ISS.EQ.0)  GO TO 1
      WRITE (6,16)
   16 FORMAT(1H1,42X,46HINPUT VALUES  X(I) WITH AVERAGE (D.C.) REMOVED)
      WRITE(6,17)
   17 FORMAT(1HT,42X,46H**********************************************)
      GO TO 2
    1 WRITE(6,6)
    6 FORMAT(1H1,42X,46HFILTERED DATA X(I) WITH AVERAGE (D.C.) REMOVED)
      WRITE(6,17)
    2 WRITE (6,5) (X(J),J=1,NOPTS)
    5 FORMAT(12X,18F6.0,7X)
      RETURN
      END
```

```
      SUBROUTINE  MAX MIN (X,AMIN,AMAX,N)
C
C     TO FIND THE MAXIMUM AND MINIMUM ELEMENT IN THE VECTOR X(I)
C
      REAL  X(10)
C
C     TO FIND THE MINIMUM ELEMENT
C
      AMIN=X(1)
      DO 10 I=1,N
      IF(X(I).LT.AMIN)  AMIN=X(I)
   10 CONTINUE
C
C     TO FIND THE MAXIMUM ELEMENT
C
      AMAX=X(1)
      DO 20 I=1,N
      IF(X(I).GT.AMAX)  AMAX=X(I)
   20 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE  FACTOR (X,NOPTS,DELT)
      REAL  X(10)
C
C     IT ADJUSTS FOR THE FACTOR DELT WHICH COMES IN WHEN WE APPROXIMATE
C     POWER FROM INTEGRAL TO SUMMATION FORM.
C
      DO 1 I=1,NOPTS
      X(I)=X(I)*DELT
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE LAGS (M, C )
C
C     THE AUTO-CORRELATIONS ARE STORED IN
C         C(T), T=1,2,...,M+1.  WHERE M IS A POWER OF 2.
C      IN ORDER TO USE THE ONE DIMENSIONAL FAST FOURIER PROGRAM,
C      IT IS NECESSARY TO HAVE THE NEGATIVE LAGS AS WELL. SINCE THE
C     AUTO-CORRELATION FUNCTION IS AN EVEN FUNCTION, AND BECAUSE OF THE
C      CYCLICAL ASPECT OF THE FAST FOURIER TRANSFORM, THE NEGATIVE LAGS
C      CAN BE STORED IN C(T), T=M+2,M+3,...,2*M. IN REVERSE ORDER.
C
      REAL  C (10)
      INTEGER M, TWO M, J, JM
C
      TWO M = 2*M
      DO 10 J=2,M
         JM = TWO M - J + 2
         C(JM) = C(J)
   10    C O N T I N U E
      R E T U R N
      E N D
```

```
      SUBROUTINE DANLAG ( N, M, C )
C
C      MODIFIES THE AUTO-CORRELATIONS C(T) BY THE DANIELL LAG WINDOW.
C         THE RESULTS ARE STORED IN C(T), T=1,2,...,N+1.
C
      REAL  C (10), PI BY M, FACTOR
      INTEGER N, M, J
C
      PI BY M = 3.141592654/FLOAT(M)
      DO 10 J=2,N
         FACTOR = PI BY M * FLOAT(J-1)
         C(J) = C(J) * SIN( FACTOR )/FACTOR
   10    C O N T I N U E
      C(N+1) = 0.0
      R E T U R N
      E N D
```

```
      SUBROUTINE COPY ( A, FIRST, LAST, STEPS, B )
C
C      COPIES THE VECTOR A INTO THE VECTOR B.
C         THE ELEMENTS OF A WHICH ARE COPIED INTO B BEGIN AT A(FIRST)
C      AND GO TO A(LAST) IN STEPS OF 'STEPS'.
C         THE ELEMENTS FROM A ARE STORED CONSECUTIVELY IN B
C      I.E.  B(I), I=1,2,...,N  WHERE N = (LAST - FIRST)/STEPS
C
C         IN THE CASE WHERE THE REMAINDER IN (LAST - FIRST)/STEPS IS NOT
C      ZERO, THE PROGRAM WILL STOP SHORT OF THE ELEMENT A(LAST).
C      I.E.  LAST IS AN UPPER BOUND, NOT A LEAST UPPER BOUND
C
      REAL  A (10), B (10)
      INTEGER FIRST, LAST, STEPS, I, J
C
      I = 0
      DO 10 J=FIRST, LAST, STEPS
         I = I + 1
         B(I) = A(J)
   10    C O N T I N U E
      R E T U R N
      E N D
```

```
      SUBROUTINE SCALE (N,X,SC)
C
      REAL X (10), SC
      INTEGER N
C
      REAL T
C
C     RESCALES (BY DIVISION) THE ARRAY X BY THE FACTOR SC FOR J=1,...,N.
C
      T=1.0/SC
      DO 100 J=1,N
      X(J)=X(J)*T
  100 CONTINUE
      RETURN
      END
```

APPENDIX B

Filter Coefficients

The theory of box type filters used in the present study have been dealt in detail in GAG report 9, 1955, and the bandpass filter Fortran program based on the same principles was published by Robinson (1966). In this filter theory, the transfer function is assumed to be an even function about the origin and consists of a unit step at the cut-off wave number. For a symmetric and zero phase shift filter, the transfer function is:

$$H(k_1) = \frac{F_0}{2} + \sum_{n=1}^{M} F_n \cos \frac{\pi n k_1}{k_{NYQ}} \qquad \text{.....(B-1)}$$

Where $\lambda_{NYQ} = \frac{2\pi}{k_{NYQ}}$ is the Nyquist wavelength for the wave number k_{NYQ} and $\lambda_1 = \frac{2\pi}{k_1}$ is the cut off wavelength for the given wave number k_1. (2M+1) are the number of terms in the filter operator.

The coefficients for various filters used, are obtained from the following expressions.

Low-pass filter coefficients

$$\left.\begin{aligned} F_0 &= \frac{k_1}{k_{NYQ}} \\ F_n &= \frac{2}{n\pi} \sin \frac{n\pi k_1}{k_{NYQ}} \end{aligned}\right| \qquad \text{.....(B-2)} \quad \text{for } n=1,2,\ldots\ldots,M$$

High-pass filter coefficients:

$$\left.\begin{aligned} F_0 &= \frac{1}{k_{NYQ}} \quad (k_{NYQ} - k_1) \\ F_n &= \frac{-2}{n\pi} \quad \sin \quad \frac{n\pi k_1}{k_{NYQ}} \end{aligned}\right\} \quad \text{for} \quad n=1,2,\ldots.,M \qquad \ldots..(B-3)$$

Band-pass filter coefficients:

$$\left.\begin{aligned} F_0 &= \frac{k_2-k_1}{k_{NYQ}} \\ F_n &= \frac{2}{n\pi} \sin \frac{n\pi k_2}{k_{NYQ}} - \frac{2}{n\pi} \sin \frac{n\pi k_1}{k_{NYQ}} \end{aligned}\right\} \text{for } n=1,2,\ldots.,M \qquad \ldots..(B-4)$$

Where k_1 and k_2 are the two different high and low cut-off wave numbers.

From the filter coefficients, one can study the impulse response and the transfer function of the filters. Two dimensional impulse response of the filter is obtained by convolving the filter operator first along the rows and then a similar operation along the columns of a matrix which has a unit spike at the centre and zero's all around it. The amplitude transfer function is obtained by taking the Fourier transform of the impulse response function and can be represented in decibels as follows:

transfer function in decibels (K_x,K_y) =

$$20 \log_{10}\left(\frac{\text{amplitude transfer function } (K_x,K_y)}{\text{max. amplitude in } (K_x,K_y) \text{ space}}\right)$$

If it is desired to modify the response of the filter, a filter coefficient may be obtained by modifying the transfer function and deriving a new impulse response.

Fortran IV programs for calculating the various filter coefficient, the impulse response and transfer function, and the convolution of two dimensional data are listed as follows:

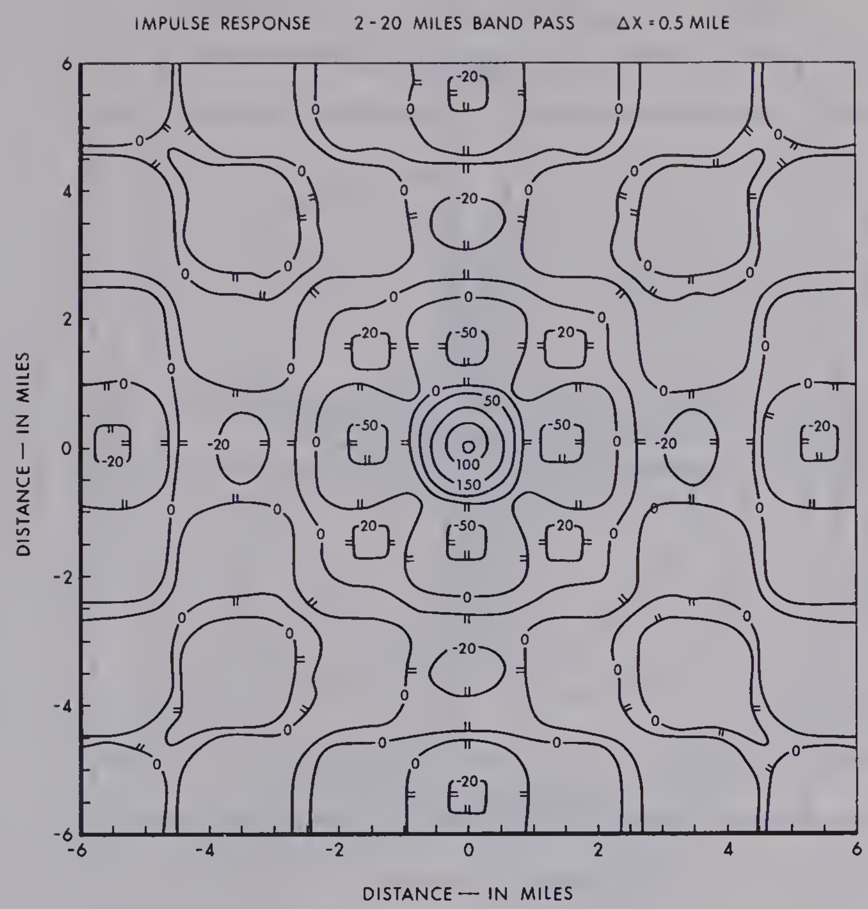


Figure B-1 (a)

Impulse response of a two-dimensional bandpass filter (2-20 miles)

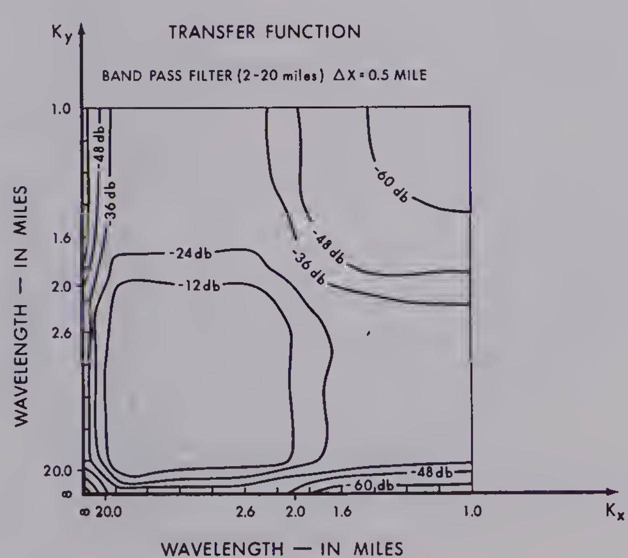


Figure B-1 (b)

NE quadrant of a transfer function for the case of figure B-1 (a)

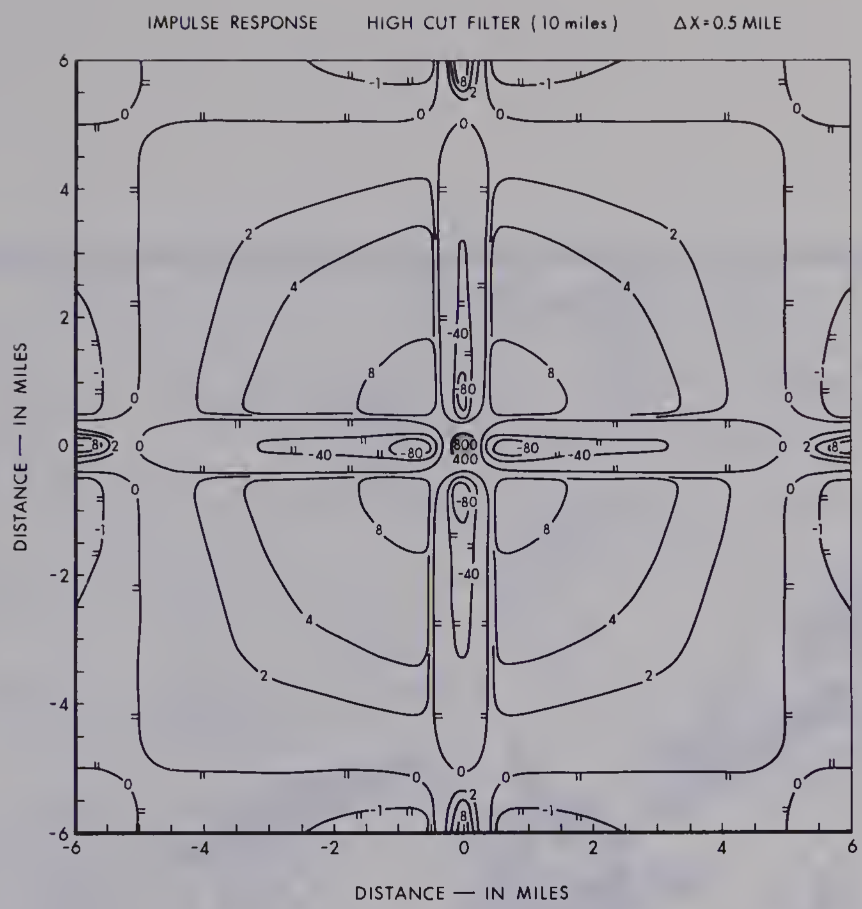


Figure B-2 (a)

Impulse response of a two-dimensional high-cut filter (which cuts off wavelengths longer than 10 miles)

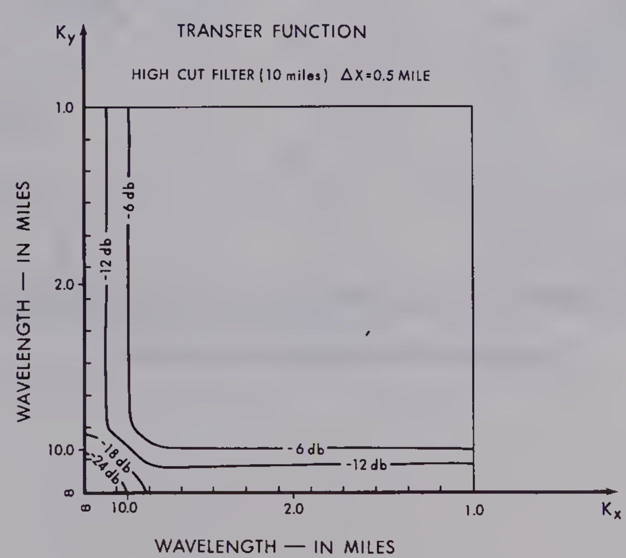
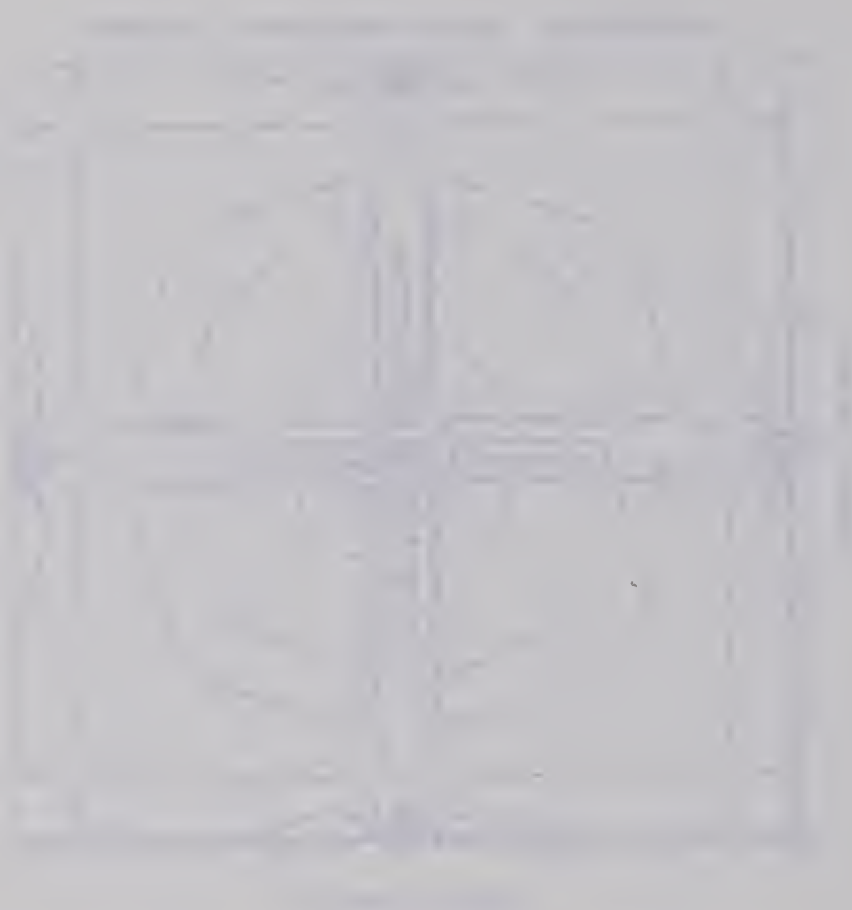


Figure B-2 (b)

NE quadrant of a transfer function for the case of figure B-2 (a)



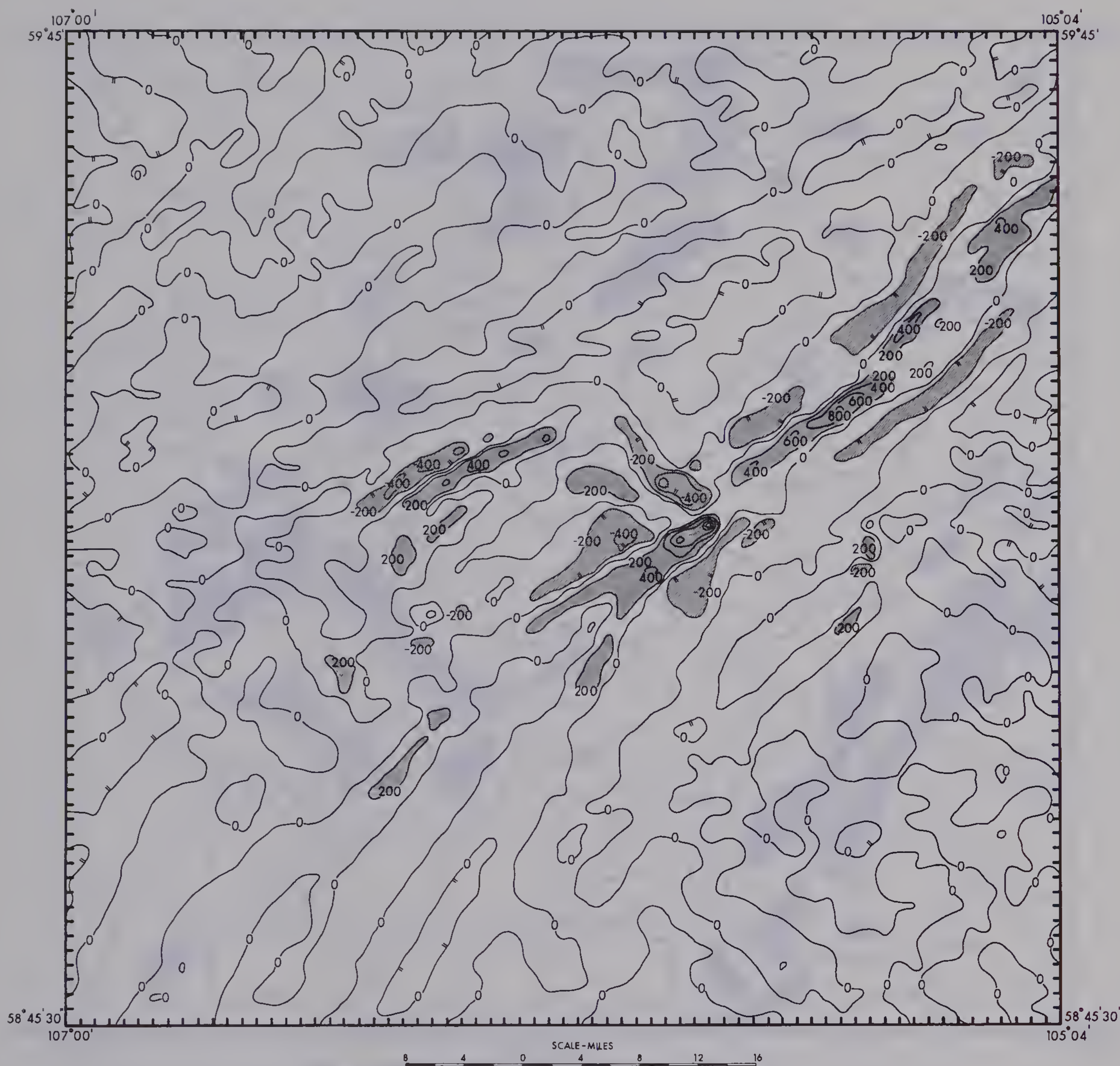


Figure B-3

Bandpass filtered magnetic map
(2-20 mile wavelengths)

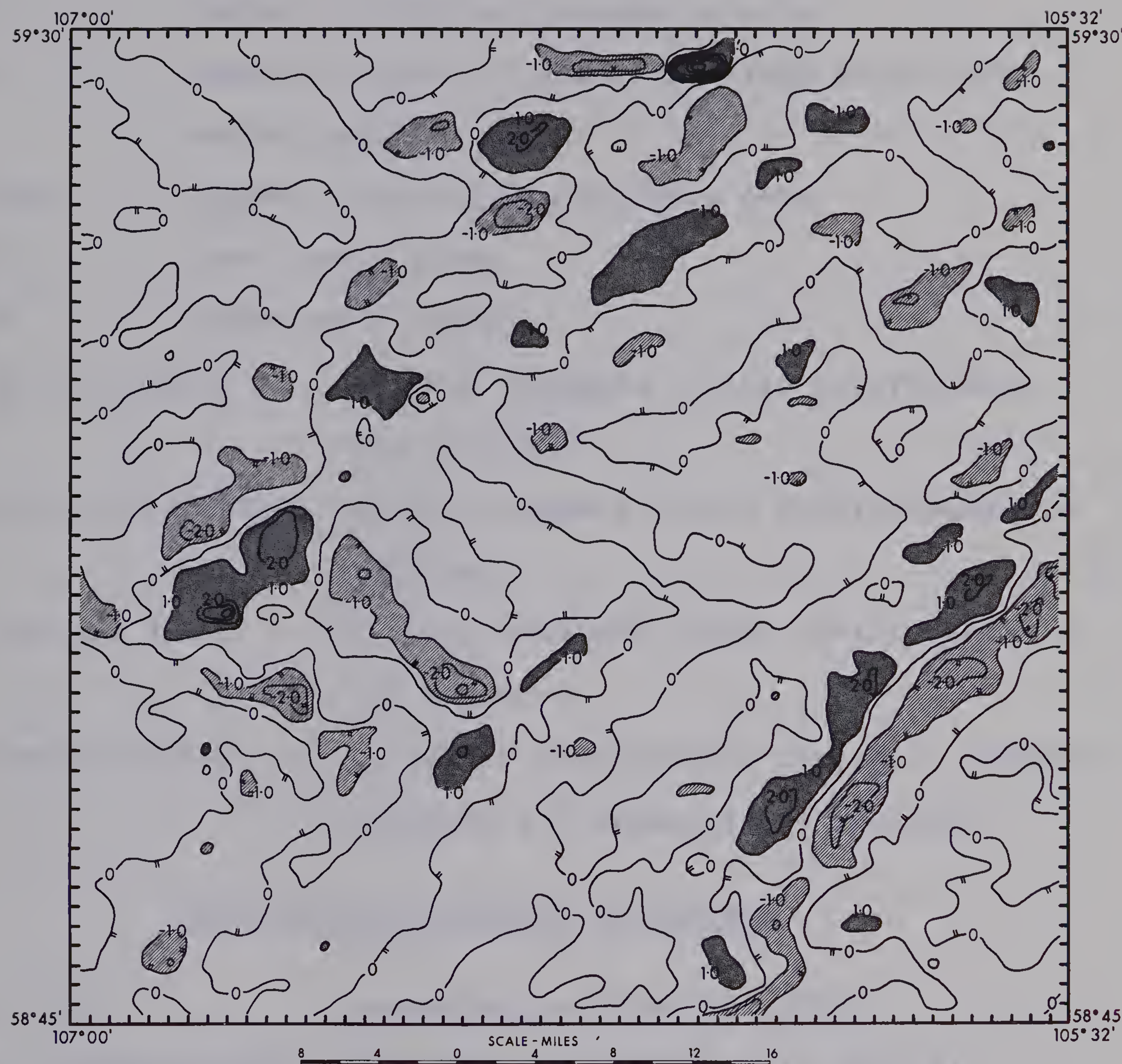
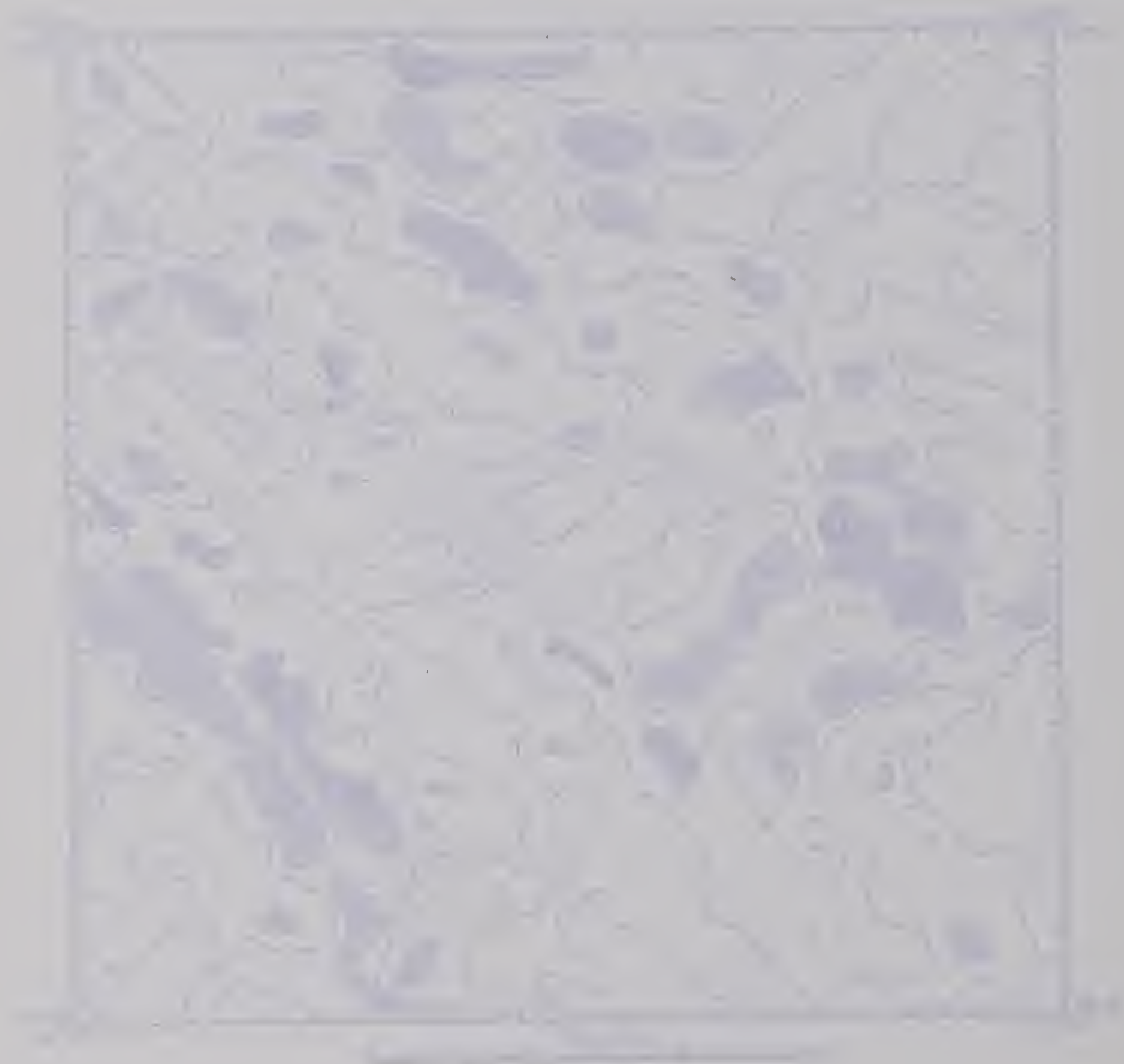


Figure B-4

Bandpass filtered gravity map
(2-20 mile wavelengths)



FORTRAN IV

PROGRAM TO OBTAIN THE FILTER COEFFICIENTS FOR HIGHPASS, LOWPASS OR BANDPASS WAVE NUMBERS USING A BOX CAR FUNCTION.

IC .... Number of different parameter sets

N .... One-half length of the symmetrical filter plus centre point

DELT ... Digital interval of the field data

FL .... Lower wave number

FH .... Higher wave number

IS=1 and ISS≠2 .... if only highpass filter coefficients are required

IS=-1 and ISS≠2.... if only lowpass filter coefficients are required

IS=0 and ISS≠2 .... if only bandpass filter coefficients are required

IS=0 and ISS=2 .... if filter coefficients for, all, lowpass, highpass and bandpass are required.

READ IN THE FOLLOWING PARAMETERS

1. IC according to FORMAT(5X,2I5)
2. N,DELT,FL,FH according to FORMAT(5X,I5,3F10.5)
3. IS,ISS according to FORMAT(5X,2I8).

```
C     PROGRAM TO OBTAIN THE FILTER COEFFICIENTS FOR HIGH PASS OR LOW PASS
C     OR BAND PASS FREQUENCIES OR TO OBTAIN THE COEFFS. OF ALL THAT IS
C     HIGH, LOW AND BAND PASS FILTERS.
C
C
C     IS= 1 AND ISS.NE.2  INDICATES THAT ONLY HIGH PASS FILTER
C                         COEFFICIENTS ARE REQUIRED.
C     IS=-1 AND ISS.NE.2  INDICATES THAT ONLY LOW PASS FILTER
C                         COEFFICIENTS ARE REQUIRED.
C     IS= 0 AND ISS.NE.2  INDICATES THAT ONLY BAND PASS FILTER
C                         COEFFICIENTS ARE REQUIRED.
C     IS= 1 AND ISS= 2    INDICATES THAT ALL FILTER COEFFICIENTS ARE
C                         REQUIRED.
C
C     IC  IS THE NUMBER OF DATA SETS TO BE READ IN.
C
C     FNYQ = NYQUIST FREQUENCY (IN CYCLES PER SECOND OR CYCLES PER MILE)
C
C     N    = LENGTH OF THE ONE LOBE OF SYMMETRICAL FILTER PLUS ONE CENTER
C             POINT.
C     FILT = THE OUTPUT FILTER OF LENGTH N.
C     FL   = THE LOWER FREQUENCY (UNITS ARE IN CYCLES PER SECONDS
C              OR CYCLES PER MILE)
C     FH   = THE HIGHER FREQUENCY (UNITS ARE IN CYCLES PER SECONDS
C              OR CYCLES PER MILE)
C     DELT = THE DITIZED INTERVAL BETWEEN DATA POINTS IN TIME OR SPACE
C              UNITS (E.G : SECONDS OR MILES ETC )
C
C
      DIMENSION   FILT(101)
      READ(5,2)  IC
      DO 300 L=1,IC
      READ(5,1)  N,DELT,FL,FH
    1 FORMAT(5X,I5,3F10.5)
      READ(5,2)  IS,ISS
    2 FORMAT(5X,2I5)
C
      M=2*N-1
      M1=M-1
      N1=N-1
      FN=N
      FNYQ=1.0/(2.0*DELT)
C        CALCULATING HIGH PASS OR LOW CUT FILTER COEFFICIENTS.
C
      IF(IS.NE.1)  GO TO 100
      CALL   HGHPAS(N,DELT,FL,FILT)
C
C     TO OBTAIN COMPLETE SET OF SYMMETRICAL FILTER COEFFICIENTS.
C
      FILT(M)=FILT(N)
      DO 25 I=N,M1
      I1=I-N1
      FILT(I)=FILT(I1)
```

```
   25 CONTINUE
      DO 26 I=1,N1
      I1=M-I+1
      FILT(I)=FILT(I1)
   26 CONTINUE
C
C          TO WRITE THE HIGH PASS FILTER COEFFICIENTS.
C
      WRITE(6,3)
    3 FORMAT(1H1)
      WRITE(6,4) N
    4 FORMAT(1HL,2X,63HLENGTH OF ONE LOBE PLUS CENTER POINT
     1              =,I5)
      WRITE(6,5)  DELT
    5 FORMAT(1HJ,2X,63HDIGITIZING INTERVAL DELT (IN TIME OR SPACE UNITS)
     1              =,F10.5)
      WRITE(6,6) FNYQ
    6 FORMAT(1HJ,2X,63HNYQUIST FREQUENCY FNYQ (IN CYCLES PER TIME OR SPA
     1CE UNITS)     =,F10.5)
      WRITE(6,7) FL
    7 FORMAT(1HJ,2X,63HLOW CUT FILTER FREQUENCY (IN CYCLES PER TIME OR S
     1PACE UNITS)  =,F10.5)
      WRITE(6,9)  M
    9 FORMAT(1HK,34X,59HCOMPLETE SET OF HIGH PASS FILTER COEFFICIENTS OF
     1 LENGTH M =,I5)
      WRITE(6,10)
   10 FORMAT(1HS,34X,64H===============================================
     1=================)
      WRITE(6,11)  (FILT(I),I=1,M)
   11 FORMAT(1X,8E16.8)
      WRITE(6,12)
   12 FORMAT(1HJ,10X,112H***********************************************
     1*****************************************************************)
C
C   CHECK IF ISS=2 , THEN WE NEED TO PASS THE CONTROL TO CALCULATE THE
C   LOW PASS FILTER COEFFICIENTS OTHERWISE WE STOP.
C
      IF(ISS.EQ.2)  GO TO 101
      GO TO 200
C
C          CALCULATING LOW PASS OR HIGH CUT FILTER COEFFICIENTS.
C
  100 IF(IS.NE.(-1))  GO TO 102
  101 CALL  LOWPAS(N,DELT,FH,FILT,IS,ISS)
C
C   TO OBTAIN COMPLETE SET OF SYMMETRICAL FILTER COEFFICIENTS.
C
      FILT(M)=FILT(N)
      DO 35 I=N,M1
      I1=I-N1
      FILT(I)=FILT(I1)
   35 CONTINUE
      DO 36 I=1,N1
```

```
      I1=M-I+1
      FILT(I)=FILT(I1)
   36 CONTINUE
C
C         TO WRITE THE LOWPASS FILTER COEFFICIENTS.
C
      IF(IS.EQ.1.AND.ISS.EQ.2)  GO TO 125
      IF(FH.LT.(1./(DELT*FN)))  GO TO 125
      WRITE(6,3)
  125 WRITE(6,4) N
      WRITE(6,15)  DELT
   15 FORMAT(1HJ,2X,63HDIGITIZING INTERVAL DELT (IN TIME OR SPACE UNITS)
     1              =,F10.5)
      WRITE(6,6) FNYQ
      WRITE(6,17) FH
   17 FORMAT(1HJ,2X,63HHIGH CUT FILTER FREQUENCY (IN CYCLES PER TIME OR
     1SPACE UNITS) =,F10.5)
      WRITE(6,19)  M
   19 FORMAT(1HK,34X,58HCOMPLETE SET OF LOW PASS FILTER COEFFICIENTS OF
     1LENGTH M =,I5)
      WRITE(6,20)
   20 FORMAT(1HS,34X,63H=================================================
     1==============)
      WRITE(6,11)  (FILT(I),I=1,M)
      WRITE(7,150)   (FILT(I),I=1,M)
  150 FORMAT(1X,4E16.8)
      WRITE(6,12)
C
C   CHECK IF ISS=2, THEN WE NEED TO PASS THE CONTROL TO CALCULATE THE
C   BAND PASS FILTER COEFFICIENTS OTHERWISE WE STOP.
C
      IF(ISS.EQ.2)  GO TO 102
      GO TO 200
C
C         CALCULATING BAND PASS FILTER COEFFICIENTS.
C
  102 IF(FL.EQ.FH)  GO TO 200
      CALL  BNDPAS(N,DELT,FL,FH,FILT,IS,ISS)
C
C   TO OBTAIN COMPLETE SET OF SYMMETRICAL FILTER COEFFICIENTS.
C
      FILT(M)=FILT(N)
      DO 45 I=N,M1
      I1=I-N1
      FILT(I)=FILT(I1)
   45 CONTINUE
      DO 46 I=1,N1
      I1=M-I+1
      FILT(I)=FILT(I1)
   46 CONTINUE
C
C         TO WRITE BAND PASS FILTER COEFFICIENTS.
C
```

```
      IF(IS.EQ.1.AND.ISS.EQ.2)  GO TO 126
      IF((FH-FL).LT.(1./(DELT*FN)))  GO TO 126
      WRITE(6,3)
  126 WRITE(6,4)  N
      WRITE(6,15)  DELT
      WRITE(6,6) FNYQ
      WRITE(6,7) FL
      WRITE(6,17) FH
      WRITE(6,21)  M
   21 FORMAT(1HK,34X,59HCOMPLETE SET OF BAND PASS FILTER COEFFICIENTS OF
     1 LENGTH M =,I5)
      WRITE(6,22)
   22 FORMAT(1HS,34X,64H================================================
     1================)
      WRITE(6,11)  (FILT(I),I=1,M)
      WRITE(6,12)
  200 CONTINUE
  300 CONTINUE
      STOP
      END
```

```
      SUBROUTINE  HGHPAS(NS,DELTS,FLS,FILTS)
C
C               HIGH PASS FILTER OR LOW FREQ. CUTOFF FILTER
C     THIS FILTER CUTS OFF THE LOW FREQ. COMPONENTS AND THE HIGH FREQ.
C     COMPONENTS ABOVE THE SPECIFIED CUT OFFFREQUENCY ARE PERMITTED TO
C     PASS THROUGH.
C
C     NS   = IS THE LENGTH OF THE ONE LOBE PLUS ONE CENTER POINT.
C     FILTS= IS THE OUTPUT FILTER OF LENGTH NS
C     FLS  = IS THE SPECIFIED FREQ. TO CUTOFF OR FILTER OUT THE
C            FREQUENCIES BELOW THE SPECIFIED ONE, WHERE AS THE FREQUENCIES
C            ABOVE FLS ARE PASSED THROUGH.
C     DELTS= DIGITIZED INTERVAL BETWEEN DATA POINTS. IN (SECONDS) OR IN
C             SPACE (MILES) . THE UNITS MENTIONED HERE ARE ARBITRARY.
C
C     IF FLS IS HIGHER THAN THE FOLDING FREQUENCY ( FLS*DELTS GREATER THAN
C     0.5) THEN THE SUBROUTINE RETURNS.
C
      DIMENSION  FILTS(101)
      IF((FLS*DELTS).GT.0.5)  GO TO 10
      FNYQ=1.0/(2.0*DELTS)
      WLS=FLS*DELTS*6.2831853
      DO 1 I=1,NS
      FILTS(I)=0.0
    1 CONTINUE
      FILTS(1)=(FNYQ-FLS)/FNYQ
      DO 2 I=2,NS
      FI=I-1
      FILTS(I)=-(SIN(WLS*FI))/FI
    2 CONTINUE
      DO 3 I=2,NS
      FILTS(I)=FILTS(I)/3.14159265
    3 CONTINUE
   10 RETURN
      END
```

```
      SUBROUTINE  LOWPAS(NS,DELTS,FHS,FILTS,IS,ISS)
C            LOW PASS FILTER OR HIGH FREQ. CUTOFF FILTER
C     THIS FILTER CUTS OFF THE HIGH FREQ. COMPONENTS AND THE LOW FRQ.
C     COMPONENTS BELOW THE SPECIFIED CUTOFF FRQ. ARE RETAINED.
C
C     NS    = IS THE LENGTH OF ONE LOBE OF A SYMMETRICAL FILTER PLUS ONE
C             CENTER POINT.
C     FILTS= IS THE OUTPUT FILTER OF LENGTH NS
C     FLS   = IS THE SPECIFIED LOWEND OF THE FREQUENCY =0.0
C             IT IS ALWAYS ZERO FOR LOW PASS FILTER.
C     FHS   = IS THE SPECIFIED FREQ. TO CUTOFF OR FILTER OUT FREQUENCIES
C             ABOVE THE SPECIFIED FHS, WHERE AS THE FREQUENCIES BELOW FHS
C             ARE RETAINED.
C     DELTS= DIGITIZED INTERVAL BETWEEN DATA POINTS, IN TIME (SECONDS) OR
C             IN SPACE (MILES). THE UNITS USED ARE ARBITRARY.
C
C     IF WE SPECIFY FLS=FHS OR (FHS-FLS=FHS BECAUSE FLS IS ALWAYS ZERO
C     IN THIS SUBROUTINE ), WE GET THE NARROWEST FILTER CONSISTENT WITH
C     THE UNCERTAINTY PRINCIPLE , CENTERED AT (FHS+FLS)/2. THAT IS FHS/2.
C     ACTUALLY IT IS NOT THE DESIRABLE FILTER AND MUST BE AVOIDED.
C
C     IF FHS IS HIGHER THAN THE FOLDING FREQUENCY (FHS*DELTS GREATER THAN
C     0.5) THEN THE SUBROUTINE RETURNS.
C
      DIMENSION  FILTS(101)
      IF((FHS*DELTS).GT.0.5)  GO TO 20
      FNYQ=1.0/(2.0*DELTS)
      FNS=NS
      IF(FHS.GE.(1./(DELTS*FNS)))  GO TO 10
      IF(IS.EQ.1.AND.ISS.EQ.2)  GO TO 6
      WRITE(6,7)
    7 FORMAT(1H1,2X,73HNARROWEST LOWPASS FILTER CENTERED AT FHS/2.0)
     1               (REJECT IT))
      GO TO 8
    6 WRITE(6,1)
    1 FORMAT(1HK,2X,73HNARROWEST LOWPASS FILTER CENTERED AT FHS/2.0)
     1               (REJECT IT))
    8 WRITE(6,2)
    2 FORMAT(1X,2X,73H
     1               ***********)
      FC=FHS/2.0
      WHS=6.2831853*(FC*DELTS+0.5/FNS)
      GO TO 15
   10 WHS=FHS*DELTS*6.2831853
   15 CONTINUE
      DO 3 I=1,NS
      FILTS(I)=0.0
    3 CONTINUE
      FILTS(1)=FHS/FNYQ
      DO 4 I=2,NS
      FI=I-1
      FILTS(I)=(SIN(WHS*FI))/FI
    4 CONTINUE
```

```
      DO 5 I=2,NS
      FILTS(I)=FILTS(I)/3.14159265
    5 CONTINUE
   20 RETURN
      END
```

```
      SUBROUTINE  BNDPAS(NS,DELTS,FLS,FHS,FILTS,IS,ISS)
C
C                           BAND PASS FILTER
C   IT PRODUCES ONE LOBE OF A SYMMETRICAL FILTER WHICH PASSES ONLY
C   FREQUENCIES BETWEEN FLS AND FHS AND PASSES THEM WITH ZERO PHASE
C   SHIFT.
C
C   NS   = IS THE LENGTH OF ONE LOBE OF A SYMMETRICAL FILTER PLUS ONE
C          CENTER POINT.
C   FILTS= IS THE OUTPUT FILTER OF LENGTH NS
C   FLS  = THE LOWER FREQUENCY.
C   FHS  = THE HIGHER FREQUENCY.
C   DELTS= DIGITIZED INTERVAL BETWEEN DATA POINTS, IN TIME (SECONDS) OR
C          IN SPACE (MILES). THE UNITS USED ARE ARBITRARY.
C
C   IF WE SPECIFY FLS=FHS OR (FHS-FLS) LESS THAN 1./(DELTS*NS), WE GET
C   THE NARROWEST FILTER CONSISTENT WITH THE UNCERTAINTY PRINCIPLE
C   CENTERED AT (FHS+FLS)/2. ACTUALLY IT IS NOT THE DESIRABLE FILTER
C   AND MUST BE AVOIDED.
C
C   IF FHS IS HIGHER THAN THE FOLDING FREQUENCY (FHS*DELTS GREATER THAN
C   0.5) THEN THE SUBROUTINE RETURNS.
C
      DIMENSION  FILTS(101)
      IF((FHS*DELTS).GT.0.5)  GO TO 20
      FNYQ=1.0/(2.0*DELTS)
      FNS=NS
      IF((FHS-FLS).GE.(1./(DELTS*FNS)))  GO TO 10
      IF(IS.EQ.1.AND.ISS.EQ.2)  GO TO 6
      WRITE(6,7)
    7 FORMAT(1H1,2X,73HNARROWEST BANDPASS FILTER CENTERED AT (FHS+FLS)/2
     1.              (REJECT IT))
      GO TO 8
    6 WRITE(6,1)
    1 FORMAT(1HK,2X,73HNARROWEST BANDPASS FILTER CENTERED AT (FHS+FLS)/2
     1.              (REJECT IT))
    8 WRITE(6,2)
    2 FORMAT(1X,2X,73H
     1              ***********)
      FC=(FHS+FLS)/2.0
      WLS=6.2831853*(FC*DELTS-0.5/FNS)
      WHS=6.2831853*(FC*DELTS+0.5/FNS)
      GO TO 15
   10 WLS=FLS*DELTS*6.2831853
      WHS=FHS*DELTS*6.2831853
   15 CONTINUE
      DO 3 I=1,NS
      FILTS(I)=0.0
    3 CONTINUE
      FILTS(1)=(FHS-FLS)/FNYQ
      DO 4 I=2,NS
      FI=I-1
      FILTS(I)=(SIN(WHS*FI)-SIN(WLS*FI))/FI
```

```
 4 CONTINUE
   DO 5 I=2,NS
   FILTS(I)=FILTS(I)/3.14159265
 5 CONTINUE
20 RETURN
   END
```

FORTRAN IV

CONVOLUTION OF TWO DIMENSIONAL DATA SET USING ONE DIMENSIONAL FILTER COEFFICIENTS.

N4	....	Size of the two dimensional data set (i.e. N4 by N4)
DELT	....	Digital interval
IS	= 1	if using highpass wave number filter coefficients
	=-1	if using lowpass wave number filter coefficients
	= 0	if using bandpass wave number filter coefficients
ISS	= 0	if only required to remove D.C. from the original data set
	= 1	if filtered data is required without decimation
	= 2	if filtered and decimated data (alternate point) is required
LF	....	Number of filter coefficient terms
FRE1	....	Low-cut wave number of the filter
FRE2	....	High-cut wave number of the filter
F(LF)	....	Vector to read in the filter coefficients
TV(N4,N4)	....	Matrix to read in the digitized data set

READ IN THE FOLLOWING PARAMETERS

1.	N4	according to FORMAT(5X,4I5)
2.	DELT	according to FORMAT(5X,F10.6)
3.	IS,ISS	according to FORMAT(5X,4I5)
4.	LF,FRE1,FRE2	according to FORMAT(5X,I5,2F10.5)
5.	F(LF)	according to FORMAT(1X,4E16.8)
6.	TV(N4,N4)	according to FORMAT(1X,6E11.4,13X)

```
C     THIS PROGRAM TAPERS THE TWO-DIMENSIONAL DATA USING COSINE FUNCTION
C     AND THEN FILTERS THE MATRIX USING ONE-DIMENSIONAL FILTER COEFFS.
C
C     THE COMPLETE SET OF INPUT DATA IS KEPT IN STORE AS A MATRIX
C     TV(I,J) OF DIMENSION N4*N4.
C
C     N4   ***   SIZE OF THE WHOLE DATA MATRIX.
C
C     DELT        ***   THE DIGITIZING INTERVAL IN MILES
C
C     FILTER COEFFS. ARE READ IN VECTOR F(200) OF KNOWN TERMS LF.
C
C     LF          ***   NUMBER OF FILTER COEFF. TERMS.
C
C     FRE1        ***   LOWCUT FREQUENCY OF THE FILTER.
C
C     FRE2        ***   HIGHCUT FREQUENCY OF THE FILTER.
C
C     WHEN USING EITHER HIGH OR LOW PASS FREQ. FILTER COEFFS; WE
C     USE FRE1=FRE2
C
C     FOR BAND PASS FILTER WE SPECIFY FRE1 AND FRE2 AS LOW
C     AND HIGH CUT FREQ. VALUES OF THE FILTER.
C
C     IS = 1      ***   MEANS USING HIGH PASS FREQ. FILTER COEFFS.
C     IS = -1     ***   MEANS USING LOW  PASS FREQ. FILTER COEFFS.
C     IS = 0      ***   MEANS USING BAND PASS FREQ. FILTER COEFFS.
C     LA, LB      ***   THE NUMBER OF POINTS AFTER CONVOLUTION.
C
C
C     ISS=0   REMOVE D.C. ONLY FROM THE ORIGINAL DATA.
C     ISS=1   OBTAIN FILTERED DATA WITHOUT DECIMATICMN.
C     ISS=2   OBTAIN FILTERED AND DECIMATED DATA.
C
C     ANY CHANGE IN COMMON CARD DIMENSIONS WOULD ALSO REQUIRE CHANGE
C     IN SUBROUTINE COMMON CARDS.
C
      DIMENSION  X(500),F(200),S(500)
      COMMON  TV(200,200)
   10 FORMAT(5X,4I5)
   16 FORMAT(59X, 'FINAL DATA' )
   17 FORMAT(1HT,58X,10(1H*))
   20 FORMAT(5X,F10.6)
   29 FORMAT(5X,I5,2F10.5)
   30 FORMAT(1X,4E16.8)
   35 FORMAT(1X,6E11.4,13X)
   45 FORMAT(1HJ)
   46 FORMAT(1H1)
   60 FORMAT(57X,'DELT  =',F8.3,'MILE',8X,'NYQUIST WAVELENGTH=',F8.3,'MI
     1LE')
   70 FORMAT(1H4)
   71  FORMAT(1H7)
   75 FORMAT(1HJ,21X, 'SIZE OF THE ORIGINAL MATRIX                    =',I5,28
```

```
     1X,'ISS = ',I5)
 76  FORMAT(1HJ,21X, 'SIZE OF DECIMATED MATRIX AFTER FILTERING =',I5,28
     1X,'ISS = ',I5)
      READ(5,10)  N4
      READ(5,20)  DELT
      READ(5,10)  IS,ISS
      READ(5,29) LF,FRE1,FRE2
      READ(5,30) (F(I),I=1,LF)
      ALAMD1=1./FRE1
      ALAMD2=1./FRE2
C
C     WE READ THE WHOLE DATA SET FROM N1 TO N2 POSITIONS IN A
C     MATRIX FORM, SO THAT WE HAVE THE REMAINING SPACE FOR
C     COSINE TAPERING PURPOSES.
C
      LF1=LF/2
      N1=LF1+1
      N2=N4+LF1
      N3=N4+2*LF1
      JDIME=200
      CALL  ZEROIJ(JDIME,JDIME)
      DO 600 I=N1,N2
      READ(5,35)  (TV(I,J),J=N1,N2)
 600  CONTINUE
C
C     NOW WE REMOVE THE D.C. VALUE FROM THE DATA.
C
      WRITE(6,46)
      CALL  DC OUT(N1,N2)
      IF(ISS.EQ.0)  GO TO 1
C     TO CALCULATE THE VALUE OF GAMMA FOR
C     TAPERING OF ROWS AND COLUMNS.
C     IT ALSL PRINTS OUT THE VALUE OF GAMMA.
C
      CALL CCNSTT(GAMMA,LF1)
C
C     NOW WE TAPER THE ROWS AND COLUMNS OF THE MATRIX TV(I,J).
C
      CALL TAPER(GAMMA,N1,N2,N3,LF1)
C
C     AGAIN WE REMOVE THE D.C. VALUE FROM THE TAPERED DATA.
C
      CALL  DC OUT(1,N3)
      WRITE( 6,70)
C     TO WRITE THE TYPE OF FILTER USED AND OTHER NECESSARY DATA
C
      CALL L H B (IS,FRE1,FRE2,ALAMD1,ALAMD2)
C
C     TO FILTER EACH ROW AND THEN TO FILTER EACH
C     COLUMN. WE USE VECTOR X(J)   FOR TRANSFERRING
C     EACH ROW AND EACH COLUMN FOR WORKING PURPOSES.
C
C     FOR ROWS
```

```
C
      DO 202 I=1,N3
      DO 203 J=1,N3
      X(J)=TV(I,J)
 203  CONTINUE
C
      CALL  FILTER(X,N3,F,LF,S,LA)
C
C     THE FILTERED VECTOR IS STORED IN S, WHICH WE TRANSFER
C     BACK TO MATRIX TV(I,J)
C
      DO 204 J=1,N3
      J1=J+LF1
      TV(I,J)=S(J1)
 204  CONTINUE
 202  CONTINUE
C
C     FOR COLUMNS
C
      DO 205 J=1,N3
      DO 206 I=1,N3
      X(I)=TV(I,J)
 206  CONTINUE
      CALL  FILTER(X,N3,F,LF,S,LA)
      DO 207 I=1,N3
      I1=I+LF1
      TV(I,J)=S(I1)
 207  CONTINUE
 205  CONTINUE
C
C     AFTER FILTERING, WE HAVE THE DATA MATRIX OF THE
C     SAME SIZE AS ORIGINALLY PUT IN. NOW WE CAN ARRANGE
C     THE MATRIX TV(I,J) AS I,J=1,N4
C
   1  CONTINUE
      DO 208 I=N1,N2
      DO 208 J=N1,N2
      I1=I-LF1
      J1=J-LF1
      TV(I1,J1)=TV(I,J)
 208  CONTINUE
C
C
C     TO PRINT THE PARAMETERS BEFORE THE DECIMATION OF DATA POINTS.
C
      WRITE(6,71)
      WRITE(6,75)  N4,ISS
      WRITE(6,45)
      FN=2.0*DELT
      WRITE(6,60)  DELT,FN
      IF(ISS.EQ.0.OR.ISS.EQ.1)  GO TO 2
C
C     WE HAVE FILTERED THE DATA BY CUTTING OFF WAVELENGTHS SMALLER THAN
```

```
C     TWO MILES. SO WE CAN DROP EVERY SECOND POINT FROM THE ORIGIONALLY
C     FILTERED DATA OF 1/4 OR 1/2 MILE DIGITIZING INTERVAL.
C
      N4=(N4+1)/2
      DO 9 I=1,N4
      I1=2*I-1
      DO 9 J=1,N4
      J1=2*J-1
      TV(I,J)=TV(I1,J1)
    9 CONTINUE
C
C     TO PRINT THE PARAMETERS AFTER THE DECIMATION OF DATA
C
      WRITE(6,45)
      WRITE(6,45)
      WRITE(6,45)
      WRITE(6,76)  N4,ISS
      WRITE(6,45)
      DELT=2.0*DELT
      FN=2.0*DELT
      WRITE(6,60)  DELT,FN
C
C     TO WRITE THE DECIMATED DATA IN THE MAP FORM
C
    2 CONTINUE
      WRITE(6,46)
      WRITE(6,16)
      WRITE(6,17)
      CALL  SPLIT(N4,N4)
      DO 15 I=1,N4
      WRITE(7,35)  (TV(I,J),J=1,N4)
   15 CONTINUE
      STOP
      END
```

```
      SUBROUTINE  ZEROIJ(N,M)
C     IT ZERO'S THE MATRIX TV(I,J)
      COMMON  TV(200,200)
      DO 1 I=1,N
      DO 1 J=1,M
      TV(I,J)=0.0
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE DC OUT(N1,N2)
C     REMOVES THE AVERAGE (D.C. ) VALUE FROM THE DATA
      COMMON  TV(200,200)
      SUM=0.0
      DO 1 I=N1,N2
      DO 1 J=N1,N2
      SUM=SUM+TV(I,J)
    1 CONTINUE
      SUM1=SUM/FLOAT((N2-N1+1)*(N2-N1+1))
      DO 2 I=N1,N2
      DO 2 J=N1,N2
      TV(I,J)=TV(I,J)-SUM1
    2 CONTINUE
C
C     TO PRINT OUT THE RESULTS
      WRITE(6,10)  SUM1
   10 FORMAT(1HJ,20X, 'AVERAGE (D.C.) VALUE    =',E13.4)
C
      RETURN
      END
```

```
      SUBROUTINE CONSTT(GAMMA,LF1)
C
C     IT CALCULATES THE CONSTANT GAMMA FOR TAPERING OF THE PROFILES
C
C     WE NEED DISTANCE FROM THE BOUNDARY  TO THE POINT
C     TO WHICH THE TAPERING IS REQUIRED.
C
C     FORMULA USED IS    GAMMA = PI/LF1
C
      GAMMA=3.14159/FLOAT(LF1)
      WRITE(6,2) GAMMA
    2 FORMAT(1H0,20X, 'TAPERING CONSTANT       =',E13.4)
      RETURN
      END
```

```
      SUBROUTINE TAPER(GAMMA,N1,N2,N,M)
C
C     TO TAPER THE PROFILES ALONG ROWS AND THEN ALONG COLUMNS
C     BY USING THE FORMULA TV(I,J)=AMPLITUDE*(0.5+0.5*COS(GAMMA*IX))
C     WHERE AMPLITUDE IS THE VALUE AT THE BOUNDARY OF THE PROFILE,
C     AND GAMMA IS THE CONSTANT DETERMINED BY SUBROUTINE CONST.
C     AND IX TAKES THE VALUE FROM 1 TO LF1=12 AT EACH END
C     OF THE PROFILE
C
      COMMON  TV(200,200)
      N31=N2+1
      DO 1 I=N1,N2
      DO 1 J=1,M
      A=N1-J
      TV(I,J)=TV(I,N1)*(0.5+0.5*COS(GAMMA*A))
      DO 1 K=N31,N
      B=K-N2
      TV(I,K)=TV(I,N2)*(0.5+0.5*COS(GAMMA*B))
    1 CONTINUE
      DO 2 I=1,N
      DO 2 J=1,M
      A=N1-J
      TV(J,I)=TV(N1,I)*(0.5+0.5*COS(GAMMA*A))
      DO 2 K=N31,N
      B=K-N2
      TV(K,I)=TV(N2,I)*(0.5+0.5*COS(GAMMA*B))
    2 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE  L H B (IS,FRE1,FRE2,ALAMD1,ALAMD2)
C
C     IT WRITES OUT LOW, HIGH, OR BAND PASS FILTER TITLES AS REQUIRED.
C
      IF(IS.EQ.1) GO TO 22
      IF(IS.EQ.0) GO TO 23
C
C    PRINT FOR LOW PASS FREQUENCY FILTER
C
      WRITE(6,60) FRE2
   60 FORMAT(1HJ,2X,29HHIGH CUT FREQUENCY (IN CPM) =,F10.5)
      WRITE(6,61) ALAMD2
   61 FORMAT(1HJ,2X,33HHERE THE WAVELENGTHS SMALLER THAN ,F5.1,1X,17HMIL
     1ES ARE CUT OFF)
      WRITE(6,62)
   62 FORMAT(1HL,35X,60HLOW PASS FREQUENCY OR LOW CUT WAVELENGTH FILTER
     1COEFFICIENTS)
      WRITE(6,63)
   63 FORMAT(1HT,35X,60H*************************************************
     1********* **)
      GO TO 41
C
C    PRINT FOR HIGH PASS FREQUENCY FILTER
C
   22 WRITE(6,70) FRE1
   70 FORMAT(1HJ,2X,29HLOW CUT FREQUENCY (IN CPM)  =,F10.5)
      WRITE(6,71) ALAMD1
   71 FORMAT(1HJ,2X,33HHERE THE WAVELENGTHS GREATER THAN ,F5.1,1X,17HMIL
     1ES ARE CUT OFF)
      WRITE(6,72)
   72 FORMAT(1HL,34X,62HHIGH PASS FREQUENCY OR HIGH CUT WAVELENGTH FILTE
     1R COEFFICIENTS)
      WRITE(6,73)
   73 FORMAT(1HT,34X,62H*************************************************
     1*********M****)
      GO TO 41
C
C    PRINT FOR BAND PASS FREQUENCY FILTER
C
   23 WRITE(6,70) FRE1
      WRITE(6,60) FRE2
      WRITE(6,81)  ALAMD2,ALAMD1
   81 FORMAT(1HJ,2X,33HHERE ONLY THE WAVELENGTHS BETWEEN,F5.1,1X,9HMILES
     1 AND,F5.1,1X,18HMILES ARE RETAINED)
      WRITE(6,82)
   82 FORMAT(1HL,50X,29HBAND PASS FILTER COEFFICIENTS)
      WRITE(6,83)
   83 FORMAT(1HT,50X,29H*****************************)
   41 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE FILTER(WORK,N3,F,LF,S,LA)
C     IT CONVOLVES THE ROW OR COLUMN VECTOR WITH FILTER COEFFS
C     AND CUTS OFF THE END EFFECTS. FINAL VECTOR IS STORED IN S
C
      REAL WORK(10),S(10),F(10)
      CALL CONVOL(WORK,N3,F,LF,S,LA)
C     TO CUT OFF THE END EFFECTS OF THE FILTERED DATA
      CALL CUTEND(S,N3,LF,LA)
      RETURN
      END
```

```
      SUBROUTINE CONVOL(X,LX,F,LF,A,LA)
C
C     EACH ROW OR COLUMN VECTOR X OF GIVEN NUMBER
C     OF POINTS LX IS CONVOLVED WITH THE FILTER COEFFS. IN
C     VECTOR F OF KNOWN NUMBER OF POINTS LF
C     THE RESULTING VALUES ARE STORED IN VECTOR A AND
C     LA GIVES TE NUMBER OF POINTS IN VECTOR A
C
      REAL X(10),F(10),A(10)
      LA=LX+LF-1
      DO 1 I=1,LA
      A(I)=0.0
    1 CONTINUE
      DO 2 I=1,LX
      DO 2 J=1,LF
      K=I+J-1
      A(K)=X(I)*F(J)+A(K)
    2 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CUTEND(X,NOPTS,LF,LA)
      REAL X(10)
C
C     IT CUTS OFF END EFFECTS OF THE FILTERED DATA
C
      LF1=LF/2
      LF2=LA-LF1+1
      DO 1 I=1,LF1
      X(I)=0.0
    1 CONTINUE
      DO 2 I=LF2,LA
      X(I)=0.0
    2 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE  SPLIT(N,M)
C
C     THIS PROGRAM SPLITS THE WRITE STATEMENT IN THE FORMAT I=1,N AND
C     J=1,26 AT A TIME.
C
C     WE CAN JOIN THE DIFFERENT WRITTEN PARTS IN THE FORM OF A MAP.
C
      COMMON  TV(200,200)
   45 FORMAT(1X,26F5.0)
   46 FORMAT(1H1)
   50 FORMAT(1H0)
      M1=M/26
      IF(M1.LT.1)  GO TO 4
      DO 1 IC=1,M1
      I1=(26*(IC-1))+1
      I2=I1+25
      DO 2 I=1,N
      WRITE(6,45)  (TV(I,J),J=I1,I2)
      WRITE(6,50)
      WRITE(6,50)
    2 CONTINUE
      WRITE(6,46)
      WRITE(6,50)
      WRITE(6,50)
    1 CONTINUE
      IF((M1*26).EQ.M)  RETURN
    4 I3=(26*M1)+1
      DO 3 I=1,N
      WRITE(6,45)  (TV(I,J),J=I3,M)
      WRITE(6,50)
      WRITE(6,50)
    3 CONTINUE
      WRITE(6,46)
      RETURN
      END
```

FORTRAN IV

TWO DIMENSIONAL IMPULSE RESPONSE AND TRANSFER FUNCTION OF THE FILTER OPERATOR.

N4	....	size of the two-dimensional coefficient matrix
N(1)		= N4-1
N(2)		= N4-1
N(3)		= 0
DELT	....	Digital interval
IS		= 1 for highpass wave number filter operators
		=-1 for lowpass wave number filter operators
		= 0 for bandpass wave number filter operators
LF	....	Number of one-dimensional filter coefficient terms
FRE1	....	Low-cut wave number of the filter
FRE2	....	High-cut wave number of the filter
F(LF)	....	Vector to read in one-dimensional filter operators

READ IN THE FOLLOWING PARAMETERS

1. N4 according to FORMAT(5X,4I5)
2. N(3) according to FORMAT(5X,4I5)
3. DELT according to FORMAT(5X,F10.6)
4. IS according to FORMAT(5X,4I5)
5. LF,FRE1,FRE2 according to FORMAT(5X,I5,2F10.5)
6. F(LF) according to FORMAT(1X,4E16.8)

```
C     THIS PROGRAM GIVES A IMPULSE RESPONSE, AND AN TRANSFER FUNCTION IN
C     FREQUENCY DOMAIN DUE TO A TWO DIMENSIONAL FILTER.
C
C     IMPULSE RESPONSE OF THE TWO DIMENSIONAL FILTER IS FORMED BY
C     CONVOLVING THE ONE DIMENSIONAL BOX TYPE FILTER COEFFICIENTS
C     WITH A 'SPIKE'; FIRST ALONG THE ROWS AND THEN ALONG THE COLUMNS.
C
C     TRANSFER FUNCTION IS OBTAINED BY USING TWO DIMENSIONAL FAST
C     FOURIER TRANSFORMS ON A SYMMETRICAL RE-ARRANGED INPUT IMPULSE
C     RESPONSE OF TWO DIMENSIONAL FILTER COEFFS.
C
C     THE PROGRAM USES TWO DIMENSIONAL FOURIER TRANSFORM WHICH IS BASED
C     ON THE ALGORITHM DEVELOPED BY GOOD (1958) AND MODIFIED BY COOLEY-
C     TUKEY (1966) AND GENTLEMAN-SANDE (1966).
C
C     N4 **** SIZE OF THE TWO-DIMENSIONAL COEFF. MATRIX.
C     N(1)=N4-1
C     N(2)=N4-1
C
C     DELT *** THE DIGITIZING INTERVAL IN MILES.
C
C     ONE DIMENSIONAL FILTER COEFFS. ARE READ IN VECTOR F(200) OF KNOWN
C     TERMS LF.
C     LF **** NUMBER OF ONE-DIMENSIONAL FILTER COEFF. TERMS.
C     FRE1 ** LOWCUT FREQUENCY OF THE FILTER.
C     FRE2 ** HIGHCUT FREQUENCY OF THE FILTER.
C
C     WHEN USING EITHER HIGH OR LOW PASS FREQ. FILTER COEFFS; WE USE
C     FRE 1 = FRE2
C
C     FOR BAND PASS FILTER WE SPECIFY FRE1 AND FRE2 AS LOW AND HIGH
C     CUT FREQ. VALUES OF THE FILTER.
C
C     IS=1 *** MEANS USING HIGH PASS FREQ. FILTER COEFFS.
C     IS=-1 *** MEANS USING LOW PASS FREQ. FILTER COEFFS.
C     IS=0 *** MEANS USING BAND PASS FREQ. FILTER COEFFS.
C     LA,LB *** THE NUMBER OF POINTS AFTER CONVOLUTION.
C
C     ANY CHANGE IN COMMON CARD DIMENSIONS WOULD ALSO REQUIRE CHANGE IN
C     SUBROUTINE COMMON CARDS.
C
      DIMENSION X(1000),Y(1000),S(1000),F(200),N(3)
      COMMON TV(101,101)
      READ(5,10) N4
      READ(5,10) (N(J),J=1,3)
      NOPTS=N(1)*N(2)
      NN=N(1)
      NNN=N(2)
   10 FORMAT(5X,4I5)
      READ(5,20) DELT
      READ(5,10) IS
      READ(5,29) LF,FRE1,FRE2
      READ(5,30) (F(I),I=1,LF)
```

```
      WNYQ=2.0*DELT
      WRITE(6,41)  DELT,WNYQ
   41 FORMAT(1HJ,2X,'DIGITIZING INTERVAL (IN MILES) =',F6.2,10X,'NYQUIST
     1 WAVELENGTH (IN MILES) =',F6.2)
   20 FORMAT(5X,F10.6)
   29 FORMAT(5X,I5,2F10.5)
   30 FORMAT(1X,4E16.8)
      ALAMD1=1./FRE1
      ALAMD2=1./FRE2
      LF1=LF/2
      N1=LF1+1
      N2=N4+LF1
      N3=N4+2*LF1
C     WE READ IN THE 'SPIKE' AT THE CENTRE AND SET REST OF THE MATRIX
C     ZERO.
      JDIME=101
      CALL ZEROIJ(JDIME,JDIME)
      TV(LF,LF)=1.0
C     TO WRITE THE TYPE OF FILTER USED AND OTHER NECESSARY DATA
      CALL LHB(IS,FRE1,FRE2,ALAMD1,ALAMD2)
C     TO CONVOLVE WITH THE FIRST ROW AND THEN TO CONVOLVE
C     WITH EACH COLUMN.
C     WE USE VECTOR X(J) FOR TRANSFERING THE ROW AND
C     EACH COLUMN FOR WORKING PURPOSES.
C
C     FOR FIRST ROW
      DO 203 J=1,N3
      X(J)=TV(LF,J)
  203 CONTINUE
C
      CALL FILTER (X,N3,F,LF,S,LA)
C     AFTER CONVOLVING AND CUTTING OFF THE END EFFECTS, THE
C     VECTOR IS STORED IN S, WHICH WE TRANSFER BACK
C     TO MATRIX TV(I,J)
      DO 204 J=1,N3
      J1=J+LF1
      TV(LF,J)=S(J1)
  204 CONTINUE
C
C     FOR COLUMNS
      DO 205 J=1,N3
      DO 206 I=1,N3
      X(I)=TV(I,J)
  206 CONTINUE
      CALL FILTER (X,N3,F,LF,S,LA)
      DO 207 I=1,N3
      I1=I+LF1
      TV(I,J)=S(I1)
  207 CONTINUE
  205 CONTINUE
C     AFTER CONVOLVING WE HAVE THE DATA MATRIX OF THE SAME
C     SIZE I.E. N3*N3 AS ORIGINALLY PUT IN.  WE CAN ARRANGE IT AS FROM
C     I,J=1,N4.
```

```
      DO 208 I=N1,N2
      DO 208 J=N1,N2
      I1=I-LF1
      J1=J-LF1
      TV(I1,J1)=TV(I,J)
 208  CONTINUE
C
      WRITE(6,31)
  31  FORMAT(1HL,43X,43HIMPULSE RESPONSE FOR TWO DIMENSIONAL FILTER)
      WRITE(6,32)
  32  FORMAT(1HT,43X,43(1H*))
      DO 209 I=1,N4
      WRITE(6,40)  (TV(I,J),J=1,N4)
      WRITE(6,45)
      WRITE(7,49)  (TV(I,J),J=1,N4)
  209 CONTINUE
  49  FORMAT(1X,6E11.4,13X)
  45  FORMAT(1HJ)
  46  FORMAT(1H1)
C
C     WE SET THE MATRIX TV IN A SUCH A SYMMETRY, SO THAT
C     THE IMPULSE RESPONSE OUTPUT IS IN THE CORRECT ORDER.
C     CONSIDER THE SE CORNER OF THE MATRIX AND PUT IT IN THE
C     NW CORNER AND THEN ARRANGE IN A SYMMETRIC MATRIX.
C     AND STORE IN THE VECTOR X FOR FOURIER TRANSFORMS.
C
C     SETTING SE CORNER OF MATRIX TV INTO NW CORNER.
      N5=(N4+1)/2
      DO 600 I=1,N5
      DO 600 J=1,N5
      I1=N5+I-1
      J1=N5+J-1
      TV(I,J)=TV(I1,J1)
 600  CONTINUE
C
C     SETTING NW CORNER (SIZE N5*N5) OF MATRIX TV IN A SYMMETRICAL ORDER
C     OF (2*N5-2)*(2*N5-2) MATRIX AND STORING IN VECTOR X.
      DO 1 I=1,N5
      DO 2 J=1,N5
      J1=(2*N5-2)*(I-1)+J
      X(J1)=TV(I,J)
   2  CONTINUE
      N6=N5-2
      DO 3 J=1,N6
      J2=J1+J
      X(J2)=TV(I,N5-J)
   3  CONTINUE
   1  CONTINUE
      N7=2*N5-2
      DO 5 K=1,N6
      DO 4 J=1,N7
      J3=J2+J
      X(J3)=X(J3-(4*N5-4)*K)
```

```
    4   CONTINUE
        J2=J3
    5   CONTINUE
C       END
C
        CALL ZERO(Y,NCPTS)
        CALL ZERO(S,NCPTS)
        CALL ARMDFT(N,X,Y,S)
C       TO WRITE THE IMPULSE RESPONSE AMPLITUDE X
C
C       NOW WE SEPARATE THE NW CORNER (N5*N5) OF THE MATRIX (N4*N4) FROM
C       VECTOR X. THEN SET NW CORNER INTO SE CORNER OF THE MATRIX (N4*N4)
C       AND COMPLETE THE SYMMETRICAL ARRANGEMENT OF THE MATRIX.
C
        DO 8 I=1,N5
        I1=N5+I-1
        DO 9 J=1,N5
        J1=N5+J-1
        J2=N5-J+1
        TV(I1,J1)=X((2*N5-2)*(I-1)+J)
        TV(I1,J2)=TV(I1,J1)
    9   CONTINUE
    8   CONTINUE
        N8=N5-1
        DO 11 I=1,N8
        DO 12 J=1,N4
        I2=N4-I+1
        TV(I,J)=TV(I2,J)
   12   CONTINUE
   11   CONTINUE
C
        WRITE(6,46)
        WRITE(6,33)
   33   FORMAT(1HL,35X,59HTRANSFER FUNCTION (IN AMPLITUDE) FOR TWO DIMENSI
       1ONAL FILTER)
        WRITE(6,34)
   34   FORMAT(1HT,35X,59(1H*))
        DO 210 I=1,N4
        WRITE(6,40)  (TV(I,J),J=1,N4)
        WRITE(6,45)
  210   CONTINUE
   40   FORMAT(1X,16F8.4)
C
C       TO NORMALIZE, WE DIVIDE ALL VALUES WITH THE HIGHEST ONE.
C       TO OBTAIN THE MAXIMUM ELEMENT FROM THE MATRIX.
        ANORM=ABS(TV(1,1))
        DO 13 I=1,N4
        DO 13 J=1,N4
        IF(ABS(TV(I,J)).GT.ANORM)  GO TO 16
        GO TO 13
   16   CONTINUE
        ANORM=ABS(TV(I,J))
   13   CONTINUE
```

```
C     DIVIDE BY ANORM TO NORMALIZE THE RESULTS.
      DO 14 I=1,N4
      DO 14 J=1,N4
      TV(I,J)=TV(I,J)/ANORM
   14 CONTINUE
C     AND TO OBTAIN THE RESULTS IN DECIBELS WE TAKE LOGRITHM OF EACH
C     NORMALIZED VALUE AND THEN MULTIPLY BY 20 (BECAUSE IT IS AMPLITUDE)
C     IF IT WERE POWER THEN WE HAVE TO MULTIPLY BY 10.
C
      DO 15 I=1,N4
      DO 15 J=1,N4
      TV(I,J)=20.0*ALOG10(ABS(TV(I,J)))
   15 CONTINUE
C
      WRITE(6,46)
      WRITE(6,35)
   35 FORMAT(1HL,35X,58HTRANSFER FUNCTION (IN DECIBELS) FOR TWO DIMENSIO
     1NAL FILTER)
      WRITE(6,34)
      CALL  SPLIT(N4,N4)
      DO 211 I=1,N5
      WRITE(6,40)  (TV(I,J),J=N5,N4)
      WRITE(6,45)
      WRITE(7,49)  (TV(I,J),J=N5,N4)
  211 CONTINUE
      STOP
      END
```

```
      SUBROUTINE  ZEROIJ(N,M)
C     IT ZERO'S THE MATRIX TV(I,J)
      COMMON TV(101,101)
      DO 1 I=1,N
      DO 1 J=1,M
      TV(I,J)=0.0
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE  L H B (IS,FRE1,FRE2,ALAMD1,ALAMD2)
C
C     IT WRITES OUT LOW, HIGH, OR BAND PASS FILTER TITLES AS REQUIRED.
C
      IF(IS.EQ.1) GO TO 22
      IF(IS.EQ.0) GO TO 23
C
C    PRINT FOR LOW PASS FREQUENCY FILTER
C
      WRITE(6,60) FRE2
   60 FORMAT(1HJ,2X,29HHIGH CUT FREQUENCY (IN CPM) =,F10.5)
      WRITE(6,61) ALAMD2
   61 FORMAT(1HJ,2X,33HHERE THE WAVELENGTHS SMALLER THAN ,F5.1,1X,17HMIL
     1ES ARE CUT OFF)
      WRITE(6,62)
   62 FORMAT(1HL,35X,60HLOW PASS FREQUENCY OR LOW CUT WAVELENGTH FILTER
     1COEFFICIENTS)
      WRITE(6,63)
   63 FORMAT(1HT,35X,60H**************************************************
     1********* **)
      GO TO 41
C
C    PRINT FOR HIGH PASS FREQUENCY FILTER
C
   22 WRITE(6,70) FRE1
   70 FORMAT(1HJ,2X,29HLOW CUT FREQUENCY (IN CPM)  =,F10.5)
      WRITE(6,71) ALAMD1
   71 FORMAT(1HJ,2X,33HHERE THE WAVELENGTHS GREATER THAN ,F5.1,1X,17HMIL
     1ES ARE CUT OFF)
      WRITE(6,72)
   72 FORMAT(1HL,34X,62HHIGH PASS FREQUENCY OR HIGH CUT WAVELENGTH FILTE
     1R COEFFICIENTS)
      WRITE(6,73)
   73 FORMAT(1HT,34X,62H**************************************************
     1*********M****)
      GO TO 41
C
C    PRINT FOR BAND PASS FREQUENCY FILTER
C
   23 WRITE(6,70) FRE1
      WRITE(6,60) FRE2
      WRITE(6,81)  ALAMD2,ALAMD1
   81 FORMAT(1HJ,2X,33HHERE ONLY THE WAVELENGTHS BETWEEN,F5.1,1X,9HMILES
     1 AND,F5.1,1X,18HMILES ARE RETAINED)
      WRITE(6,82)
   82 FORMAT(1HL,50X,29HBAND PASS FILTER COEFFICIENTS)
      WRITE(6,83)
   83 FORMAT(1HT,50X,29H*****************************)
   41 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE FILTER(WORK,N3,F,LF,S,LA)
C     IT CONVOLVES THE ROW OR COLUMN VECTOR WITH FILTER COEFFS
C     AND CUTS OFF THE END EFFECTS. FINAL VECTOR IS STORED IN S
C
      REAL WORK(10),S(10),F(10)
      CALL CONVCL(WORK,N3,F,LF,S,LA)
C     TO CUT OFF THE END EFFECTS OF THE FILTERED DATA
      CALL CUTEND(S,N3,LF,LA)
      RETURN
      END
```

```
      SUBROUTINE CONVOL(X,LX,F,LF,A,LA)
C
C     EACH ROW OR COLUMN VECTOR X OF GIVEN NUMBER
C     OF POINTS LX IS CONVOLVED WITH THE FILTER COEFFS. IN
C     VECTOR F OF KNOWN NUMBER OF POINTS LF
C     THE RESULTING VALUES ARE STORED IN VECTOR A AND
C     LA GIVES TE NUMBER OF POINTS IN VECTOR A
C
      REAL X(10),F(10),A(10)
      LA=LX+LF-1
      DO 1 I=1,LA
      A(I)=0.0
    1 CONTINUE
      DO 2 I=1,LX
      DO 2 J=1,LF
      K=I+J-1
      A(K)=X(I)*F(J)+A(K)
    2 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CUTEND(X,NOPTS,LF,LA)
      REAL X(10)
C
C     IT CUTS OFF END EFFECTS OF THE FILTERED DATA
C
      LF1=LF/2
      LF2=LA-LF1+1
      DO 1 I=1,LF1
      X(I)=0.0
    1 CONTINUE
      DO 2 I=LF2,LA
      X(I)=0.0
    2 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE  SPLIT(N,M)
C
C     THIS PROGRAM SPLITS THE WRITE STATEMENT IN THE FORMAT I=1,N AND
C     J=1,26 AT A TIME.
C
C     WE CAN JOIN THE DIFFERENT WRITTEN PARTS IN THE FORM OF A MAP.
C
      COMMON TV(101,101)
   45 FORMAT(1X,26F5.1)
   46 FORMAT(1H1)
   50 FORMAT(1HJ)
      M1=M/26
      IF(M1.LT.1)  GO TO 4
      DO 1 IC=1,M1
      I1=(26*(IC-1))+1
      I2=I1+25
      DO 2 I=1,N
      WRITE(6,45)  (TV(I,J),J=I1,I2)
      WRITE(6,50)
      WRITE(6,50)
    2 CONTINUE
      WRITE(6,46)
      WRITE(6,50)
      WRITE(6,50)
    1 CONTINUE
      IF((M1*26).EQ.M)  RETURN
    4 I3=(26*M1)+1
      DO 3 I=1,N
      WRITE(6,45)  (TV(I,J),J=I3,M)
      WRITE(6,50)
      WRITE(6,50)
    3 CONTINUE
      WRITE(6,46)
      RETURN
      END
```

```
      SUBROUTINE ZERO(X,NOPTS)
C     SETS THE VECTOR TO ZERO
      REAL X(10)
      DO 1 J=1,NOPTS
      X(J)=0.0
    1 CONTINUE
      RETURN
      END
```

APPENDIX C

Continuation and derivative processes

The continuation and derivative operations are performed by using the fast FOURIER transform algorithm. It must be noted that these processes are possible only if the original potential field data is arranged according to the cyclic properties of the FOURIER transforms. For example, the actual physical data $A(i,j)$ (for $i,j=1,2,\ldots,n$) must be set in such a way that the analysis is performed on the matrix $x(i,j)$ (for $i,j=1,2,\ldots,2n-2$):

(a) $x(i,j) = A(i,j)$ for $i,j=1,2,\ldots.n$

(b) $x(i,n+k) = A(i,n-k)$ for $i=1,2,\ldots n$ and $k=1,2,\ldots,(n-2)$

(c) $x(n+\ell,j) = A(n-\ell,j)$ for $j=1,2,\ldots n$ and $\ell=1,2,\ldots,(n-2)$

(d) $x(n+\ell,n+k) = A(n-\ell,n-k)$ for $k,\ell=1,2,\ldots.,(n-2)$

The flow diagram and the pictorial representation of the actual data of 4 by 4 size (figure C-1 and C-2) would clarify the idea further.

The Fortran IV programs for upward or downward continuations and derivatives using FFT algorithm are listed as follows:

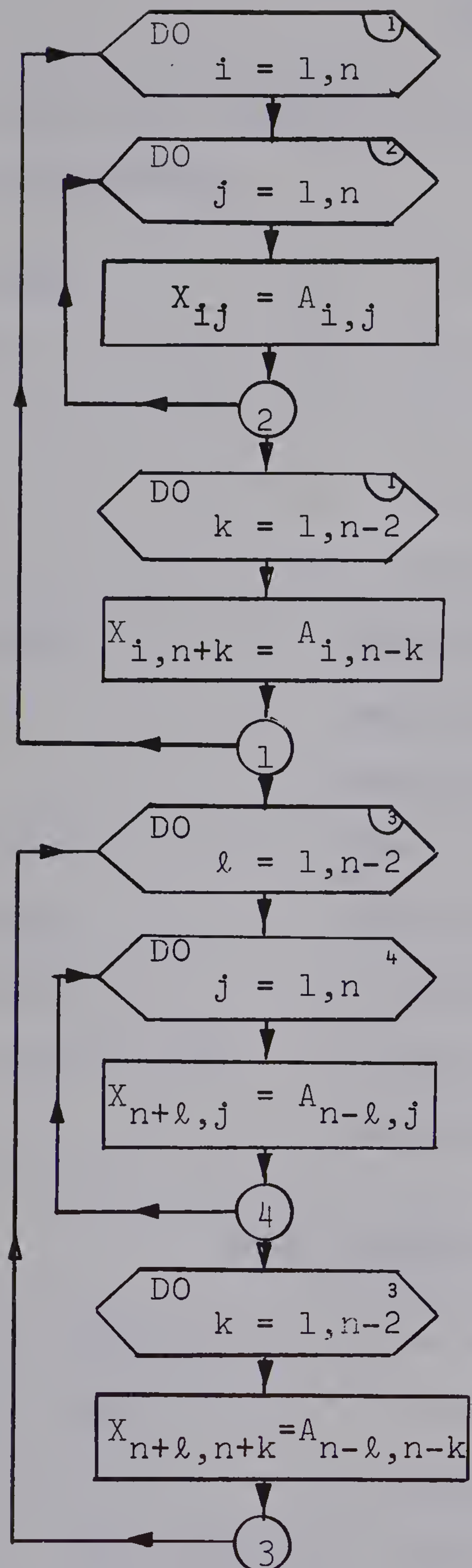


Figure C-1. Flow diagram to set up the original data $A_{i,j}$ into $X_{i,j}$ to obtain the continuation and derivatives using fast Fourier transforms.

A_{11}	A_{12}	A_{13}	A_{14}
A_{21}	A_{22}	A_{23}	A_{24}
A_{31}	A_{32}	A_{33}	A_{34}
A_{41}	A_{42}	A_{43}	A_{44}

(a)

Original data $A_{i,j}$

A_{11}	A_{12}	A_{13}	A_{14}	A_{13}	A_{12}
A_{21}	A_{22}	A_{23}	A_{24}	A_{23}	A_{22}
A_{31}	A_{32}	A_{33}	A_{34}	A_{33}	A_{32}
A_{41}	A_{42}	A_{43}	A_{44}	A_{43}	A_{42}
A_{31}	A_{32}	A_{33}	A_{34}	A_{33}	A_{32}
A_{21}	A_{22}	A_{23}	A_{24}	A_{23}	A_{22}

(b)

Re-arranged data $X_{i,j}$

Figure C-2. Pictorial representation of $A_{i,j}$ in (a) into $X_{i,j}$ in (b) for obtaining the continuation and derivatives by using F.F.T.

FORTRAN IV

UPWARD AND DOWNWARD CONTINUATION AND CALCULATION OF DERIVATIVES.

INPUT	= 5	if data is read in from the cards
ID	= 0	if no derivative is required
	= 1	if 1st. derivative is required
	= 2	if 2nd derivative is required
	= N	if Nth derivative is required
DELT		Digitizing interval of the data set
Z	=	Negative (elevation) for upward continuation
	=	Positive (elevation) for downward continuation
N(1)		Size of the data set matrix along x
N(2)		Size of the data set matrix along y
N(3)	= 0	for third dimension
X1(N(1),N(2))		Is the dimensional matrix to read in the two dimensional data set.

READ IN THE FOLLOWING PARAMETERS

1. INPUT,ID according to FORMAT(5X,3I5)
2. DELT according to FORMAT(5X,F10.5)
3. Z according to FORMAT(5X,F10.5)
4. (N(J),J=1,3) according to FORMAT(5X,3I5)
5. X1(N(1),N(2)) according to FORMAT(1X,6E11.4)

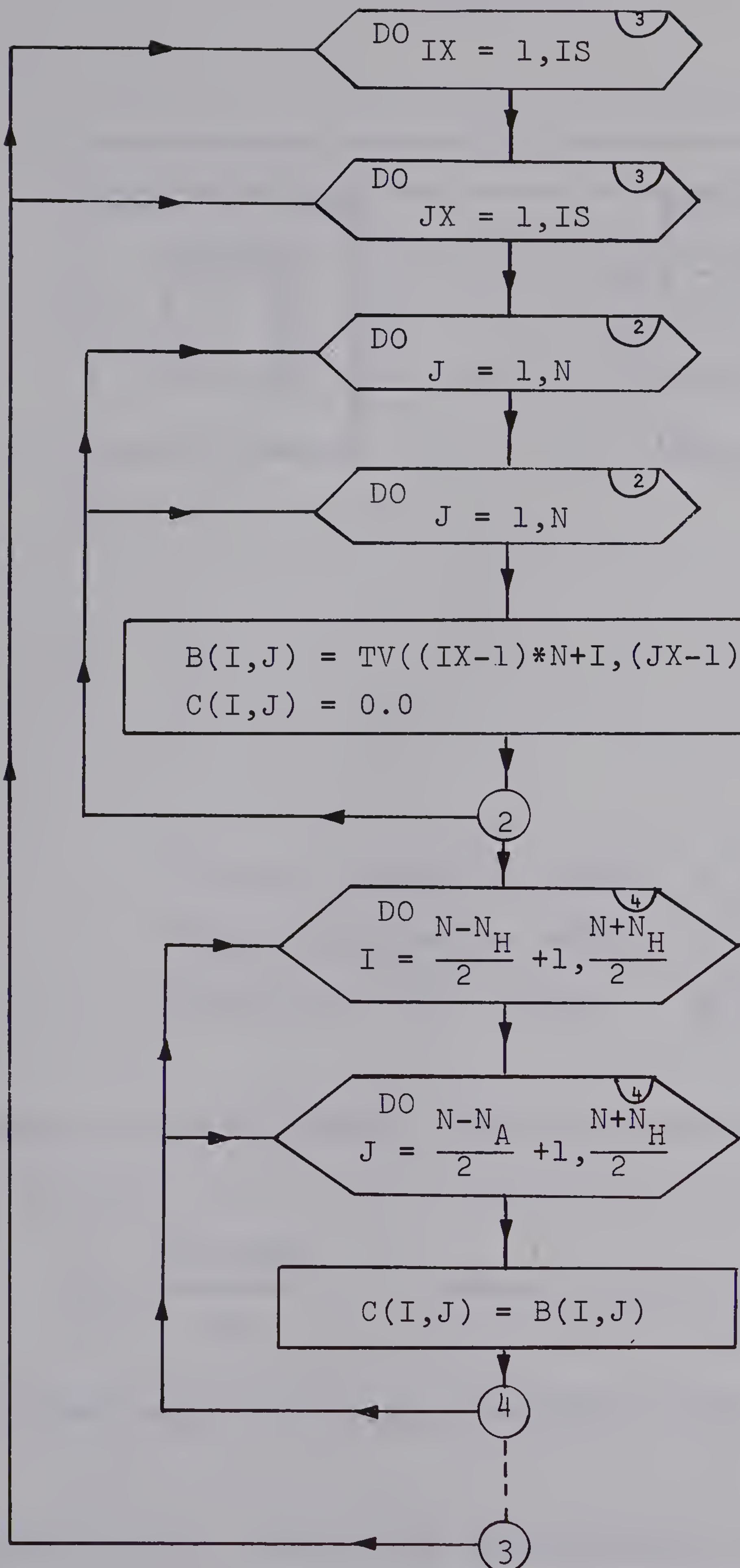
```
C
C     THIS PROGRAM CALCULATES THE UPWARD OR DOWNWARD CONTINUATION AT ANY
C     HEIGHT OR DEPTH OF THE GIVEN DATA AT A PARTICULAR LOCATION.
C
C     THIS PROGRAM ALSO OBTAINS 1ST, 2ND OR HIGHEST SET OF DERIVATIVES.
C
C     THE GAUSSION DISTRIBUTION FOR THE CALCULATIONS USED IN THIS
C     PROGRAM IS AFTER TSUBOI (1959).
C
C     THE PROGRAM USES TWO DIMENSIONAL FOURIER TRANSFORM WHICH IS BASED
C     ON THE ALGORITHM DEVELOPED BY GOOD (1958) AND MODIFIED BY COOLEY-
C     TUKEY (1966) AND GENTLEMAN-SANDE (1966).
C
C     Z  **  UPWARD (-IVE) OR DOWNWARD (+IVE) HEIGHT.
C
C     ID=0 FOR NO DERIVATIVE
C     ID=1 FOR 1ST DERIVATIVE
C     ID=2 FOR 2ND DERIVATIVE
C     ID=3 FOR 3RD DERIVATIVE AND SO ON
C
C     X1(I,J) IS THE REAL INPUT MATRIX IN SPACE DOMAIN.
C
C     X(J) IS THE FINAL SYMMETERIZED DATA, USING GYCLIC PROPORTIES OF
C     FOURIER TRANSFORMS.
C
C     FOR COMPUTATION PROGRAM USES X(J)+IY(J) WHERE Y(J)=0.0
C     OUTPUT RESULTS ARE GIVEN AS X(J)+IY(J) IN FREQUENCY DOMAIN
C     N(J,K,O)  ***  ARRAY OF DATA IS TWO DIMENSIONAL WITH SIDE J BY K.
C     NOPTS    ***    NUMBER OF DATA POINTS  (J*K).
C     DELT    ***    THE DIGITIZING INTERVAL IN MILES.
C
      DIMENSION  X(15000),Y(15000),S(15000),N(3),X1(60,60)
   20 FORMAT (5X,3I5)
   22 FORMAT(1HJ,31X,23HDEGREE OF DERIVATIVE = ,I5,23X,24HSIZE OF OUTPUT
     1 MATRIX = ,I5)
   23 FORMAT(1HJ,20X,34HELEVATION OF CONTINUATION FIELD = ,F10.5,5X,37HD
     1IGITIZING INTERVAL OF DATA POINTS = ,F10.5)
   30 FORMAT (5X,F10.5)
   35 FORMAT (1X,6E11.4)
   36 FORMAT(1X,10E13.4)
   45 FORMAT (1HJ)
   40 FORMAT(1H1,60X,10HINPUT DATA)
   42 FORMAT(1HT,60X,10H**********)
   50 FORMAT(1HT,25X,80(1H*))
   60 FORMAT(1X,6HDELT =,F8.3,8X,18HNYQUIST FREQUENCY=,F8.3)
   70 FORMAT(1HJ,22X,15HFREQUENCY =2*J*,F10.6,1X,15HCYCLES PER MILE)
  316 FORMAT(1H1,40X,47HUPWARD OR DOWNWARD CONTINUATION AND DERIVATIVES)
  317 FORMAT(41X,47(1H*))
  320 FORMAT(1HL,56X,17HREAL (SPACE) PART)
  325 FORMAT(1HT,56X,17H*****************)
      READ (5,20) INPUT,ID
      READ (5,30) DELT
      READ(5,30)  Z
```

```
      READ (5,20) (N(J),J=1,3)
      NOPTS = N(1)*N(2)
      NN=N(1)
      NNN=N(2)
      DO 9 I=1,NN
      READ(INPUT,35) (X1(I,J),J=1,NNN)
    9 CONTINUE
      WRITE (6,40)
      WRITE(6,42)
      DO 10 I=1,NN
      WRITE(6,36) (X1(I,J),J=1,NNN)
      WRITE (6,45)
   10 CONTINUE
      WRITE (6,50)
C
C     SETTING X AS THE FINAL SYMMETERIZED MATRIX OF SIZE (2N-2)*(2N-2)
C     REMEMBER HERE X IS STORED IN A VECTOR FORM.
      DO 1 I=1,NN
      DO 2 J=1,NNN
      J1=(2*NN-2)*(I-1)+J
      X(J1)=X1(I,J)
    2 CONTINUE
      N1=NN-2
      DO 3 J=1,N1
      J2=J1+J
      X(J2)=X1(I,NN-J)
    3 CONTINUE
    1 CONTINUE
      N2=2*NN-2
      DO 5 K=1,N1
      DO 4 J=1,N2
      J3=J2+J
      X(J3)=X(J3-(4*NN-4)*K)
    4 CONTINUE
      J2=J3
    5 CONTINUE
C     END OF SYMMETERIZED VECTOR.
C     NOW WE RESET N(1),N(2),NOPTS,ETC.
      N(1)=2*NN-2
      N(2)=2*NNN-2
      NN=N(1)
      NNN=N(2)
      NOPTS=N(1)*N(2)
      FN = 1./(2.*DELT)
      WRITE (6,60) DELT,FN
      FP = FN/(FLOAT(NOPTS))
      WRITE (6,70) FP
      CALL  ZERO(Y,NOPTS)
      CALL  ZERO(S,NOPTS)
      CALL  ARMDFT(N,X,Y,S)
C
C     TO MULTIPLY FREQUENCIES BY A FACTOR
C     EXP((I**2+J**2)*(2.0*PI*Z/(NN*DELT))), FOR I=0,2,2,.....NN AND
```

```
C      J=0,1,2,.....NNN
C
       DO 121 I=1,NN
       IF(I.GT.(NN/2+1)) GO TO 11
       A=I-1
       GO TO 12
    11 A=NN+1-I
    12 CONTINUE
       K=(I-1)*NNN+1
       L=K+NNN-1
       DO 121 J=K,L
       IF(J.GT.((L+K)/2+1)) GO TO 13
       B=J-1-(NN*(I-1))
       GO TO 14
    13 B=NN+1-J+(NN*(I-1))
    14 CONTINUE
       CONT=EXP((SQRT(A**2+B**2)*2.0*3.14159*Z)/(FLOAT(NN)*DELT))
       IF(ID.EQ.0) GO TO 15
       CONTD1=(SQRT(A**2+B**2)*(2.0*3.14159)/(FLOAT(NN)*DELT))**ID
    15 CONTINUE
       X(J)=X(J)*CONT
       Y(J)=Y(J)*CONT
       IF(ID.EQ.0) GO TO 121
       X(J)=X(J)*CONTD1
       Y(J)=Y(J)*CONTD1
   121 CONTINUE
       J=0
       DO 250 J3=1,NOPTS
       J=J+1
C      REPLACE FOURIER COEFFICIENTS BY COMPLEX  CONJUGATE.
       F=Y(J)
       Y(J)=-F
   250 CONTINUE
       CALL  ZERO(S,NOPTS)
       CALL  ARMDFT(N,X,Y,S)
       J=0
       DO 350 J4=1,NOPTS
       J=J+1
C      TAKE COMPLEX CONJUGATE.
       GG=Y(J)
       Y(J)=-GG
   350 CONTINUE
       J=0
       DO 400 J5=1,NOPTS
       J=J+1
       X(J)=X(J)/FLOAT(NOPTS)
       Y(J)=Y(J)/FLOAT(NOPTS)
   400 CONTINUE
C
C      RESET NN AND NNN . AND PICK UP USEFUL SIZE OF THE FINAL RESULTS.
C      WE ALSO OBTAIN PUNCHED DECK OF THE FINAL RESULTS.
C
       WRITE(6,316)
```

```
      WRITE(6,317)
      WRITE (6,320)
      WRITE (6,325)
      NN=(NN+2)/2
      NNN=(NNN+2)/2
      DO 7 I=1,NN
      DO 8 J=1,NNN
      X1(I,J)=X((2*NN-2)*(I-1)+J)
    8 CONTINUE
    7 CONTINUE
      DO 21 I=1,NN
      WRITE(6,36) (X1(I,J),J=1,NNN)
      WRITE(7,35) (X1(I,J),J=1,NNN)
      WRITE(6,45)
   21 CONTINUE
      WRITE (6,50)
      WRITE(6,45)
      WRITE(6,45)
      WRITE(6,23) Z,DELT
      WRITE(6,45)
      WRITE(6,22)ID,NN
      STOP
      END
```

```
      SUBROUTINE ZERO(X,NCPTS)
C     SETS THE VECTOR TO ZERO
      REAL X(10)
      DO 1 J=1,NCPTS
      X(J)=0.0
    1 CONTINUE
      RETURN
      END
```



IX moves the test map along the rows from 1 to IS.

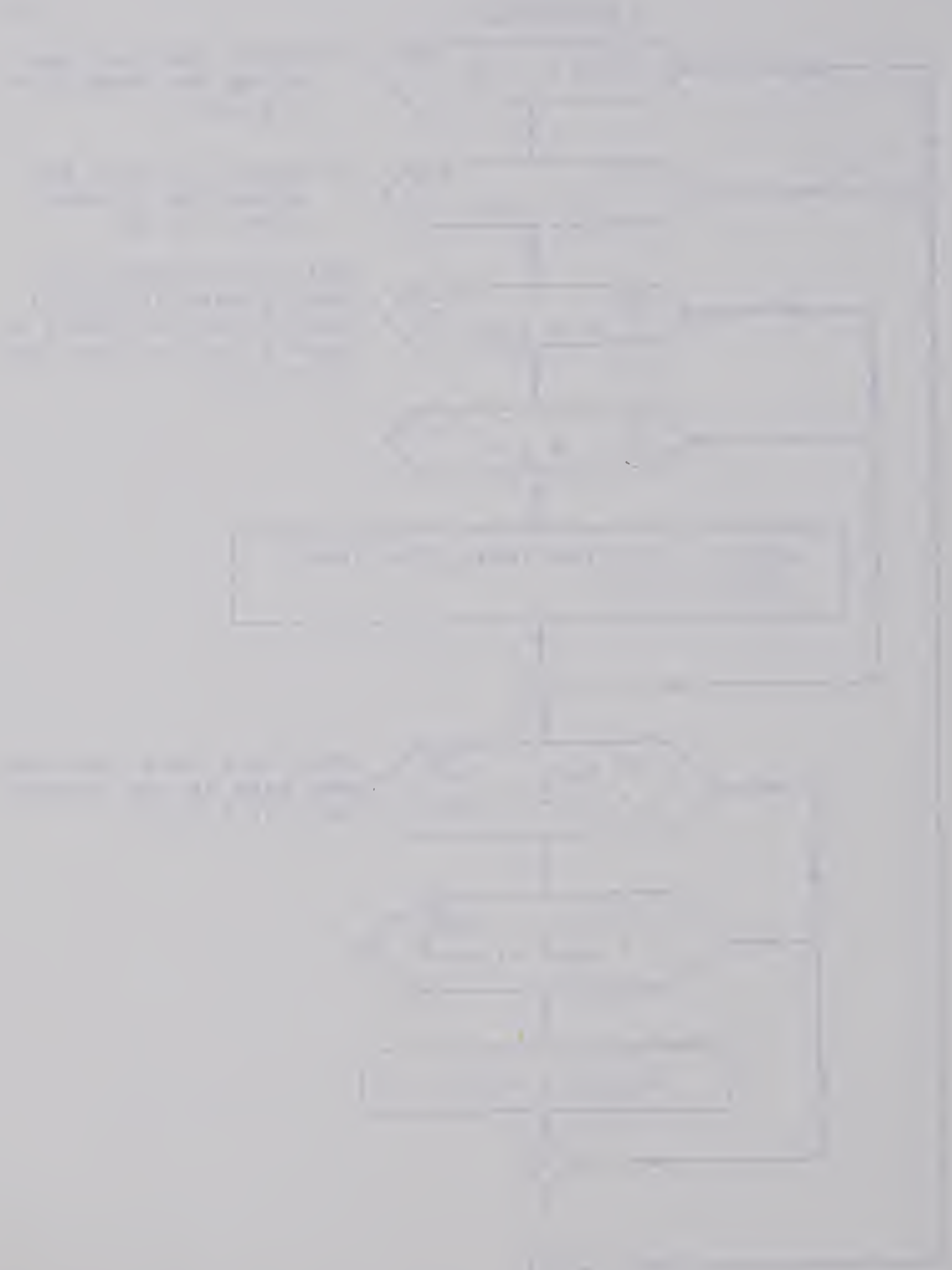
JX moves the test map along the columns from 1 to IS.

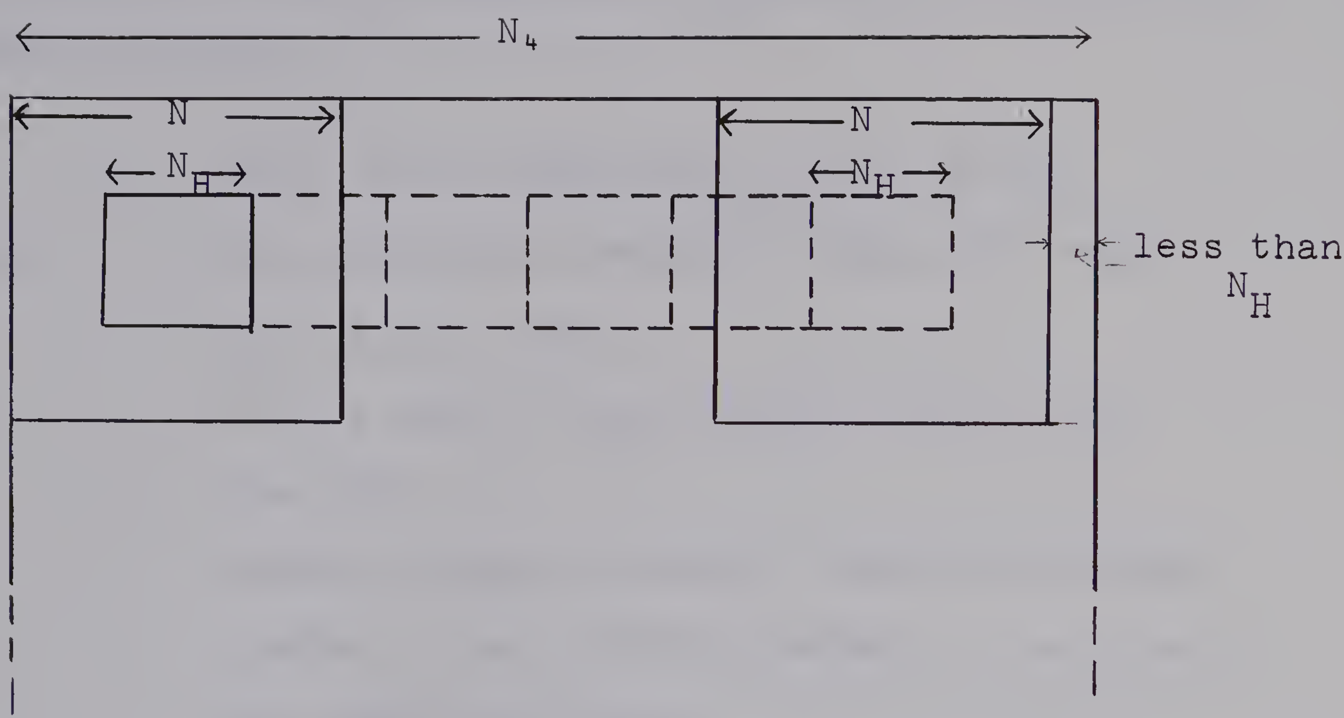
This loop places the sample data in B(I,J) and places the field of zero's for the test map in C(I,J).

This loop sets the test map data in the central core of C(I,J).

Figure (D-1)

Flow diagram for setting up the sample map and test map data in B(I,J) and C(I,J) respectively, within the displacement loop 3 to cover the entire field data.





Size of the entire matrix N_4 by N_4

Size of the sample matrix N by N

Size of the test matrix N_H by N_H

Number of steps required along each row and columns given by IS i.e.

$$IS = \frac{N_4 - N + N_H}{N_H} \qquad \text{where } IS, N_4, N, N_H \text{ are integers}$$

Consider only the integer division of IS.

Figure (D-2). Set up for displacing the test map for sampling of cross-correlations at representative sites over the entire map.

FORTRAN IV

TWO DIMENSIONAL CROSS-CORRELATION COEFFICIENT CALCULATIONS FOR TREND ANALYSIS.

N4	....	Size of the data matrix of N4 by N4.
N(J,K,0)	..	Two dimensional array of sample map data with J by K size
NHALF	....	is the size of the central core of the test map
NTAO	....	Number of lags, normally taken the maximum number of lags which is equal to the size of the sample map data.
DELT	....	Digital interval of the data set
ISS	= 1	if unfiltered data is used to obtain the cross-correlation coefficients
	= 0	if filtered data is used
TV(N4,N4)	..	Dimensioned matrix to read in the complete data set

READ IN THE FOLLOWING PARAMETERS

1. N4 according to FORMAT(5X,4I5)
2. N(3) according to FORMAT(5X,4I5)
3. NHALF according to FORMAT(5X,4I5)
4. NTAO according to FORMAT(5X,4I5)
5. DELT according to FORMAT(5X,F10.6)
6. ISS according to FORMAT(5X,4I5)
7. TV(N4,N4) according to FORMAT(1X,6E11.4)

```
C       THIS PROGRAM OBTAINS THE TWO-DIMENSIONAL CROSS-CORRELATION
C       OF THE GIVEN DATA.
C
C       THE PROGRAM MAKES USE OF THE TWO-DIMENSIONAL FOURIER TRANSFORMS,
C       BASED ON THE ALGORITHM OF GOOD(1958) AS MODIFIED BY COOLEY-TUKEY
C       (1966) AND GENTLEMAN-SANDE(1966)
C
C       FOR CROSS-CORRELATION, FIRST SET OF DATA IS READ IN 'BY' VECTOR
C       WHICH CONSISTS OF SIZE N(1)*N(2), AND THE DATA WITH WHICH IT IS
C       TO BE CORRELATED IS ENTERED IN VECTOR 'DY' WHICH IS OF THE SAME
C       SIZE AS 'BY'
C
C       N(J,K,C)    ***   ARRAY OF DATA IS TWO-DIMENSIONAL WITH SIZE J BY K
C
C       NHALF IS THE VALUE OF THE SIZE OF CENTRAL MATRIX WHICH IS TO BE
C       CROSS_CORRELATED WITH THE LARGER MATRIX OF SIZE NN.
C
C       NHALF IS HALF THE VALUE OF NN IN THE CASE OF EVEN NN.
C
C       NHALF IS ODD VALUE IN THE CASE OF ODD NN.
C
C       FOR CONVENIENCE IT IS PROPOSED TO CHOOSE NN AS ODD.
C
C       (NN-NHALF)/2=(NHALF-1)  SHOULD GIVE NN FOR A PARTICULAR GIVEN
C       VALUE OF NHALF.
C
C       N4    ***   SIZE OF THE DATA MATRIX
C
C       DELT     ***   THE DIGITIZING INTERVAL IN MILES
C
C       NOPTS        ***   NUMBER OF DATA POINTS (J*K)=N(1)*N(2)
C
C       N TAG        ***   NUMBER OF LAGS THAT ALSO MEANS THAT WE ADD
C                          SO MANY ZEROS. MAXIMUM LAGS CAN BE EQUAL TO
C                          THE DIMENSION N(1) OR N(2) OF THE DATA POINTS
C                          READ IN.
C       ISS = 1      ***   CORRELATION AND POWER SPECTRA ARE OBTAINED ON
C                          UNFILTERED DATA
C       ISS = 0      ***   CORRELATION AND POWER SPECTRA ARE OBTAINED ON
C                          FILTERED DATA
C
        DIMENSION  WORK1(50,50),WORK2(50,50),TV(104,104),TEMP(104,104)
C
C       ANY CHANGE IN TV(104,104) AND TEMP(104,104) DIMENSIONS NEED OTHER
C       CHANGE IN THE MAIN PROGRAM, THAT IS, JDIME=104 CARD IS TO BE
C       CHANGED.
C       ALSO MATRIX DIMENSIONS IN THE SUBROUTINES WOULD NEED CHANGES.
C
C       ANY CHANGE OF DIMENSIONS IN THE COMMON CARDS ALSO GOES IN THE
C       SUBROUTINE 'CROSS'. IN SUBROUTINE 'CROSS' CARD IDIME=2500 WOULD
C       ALSO NEED TO BE CHANGED.
C
        COMMON AX(2500),BY(2500),CX(2500),DY(2500),S(2500),X(2500),Y(2500)
```

```
      COMMCN  DELT
      COMMCN  N(3)
      COMMCN  NN,NNN,NOPTS,NTAO,NHALF
      REAL  AX,BY,CX,DY,S,X,Y,DELT
      INTEGER  N,NN,NNN,NCPTS,NTAO
    7 FORMAT( 46X,'CROSS_CORRELATION OF THE INPUT DATA'  )
    8 FORMAT(1HT,45X,35(1H*))
   10 FORMAT(5X,4I5)
   20 FORMAT(5X,F10.6)
   29 FORMAT(5X,I5,2F10.5)
   30 FORMAT(1X,4E16.8)
   35 FORMAT(1X,6E11.4)
   36 FORMAT(1X,10E13.4)
   45 FORMAT(1HJ)
   46 FORMAT(1H1)
   50 FORMAT(1HT,25X,80(1H*))
   60 FORMAT(22X,6HDELT =,F8.3,8X,19HNYQUIST WAVELENGTH=,F8.3)
   70 FORMAT(1HT,21X,'ISS =',I5)
   73 FORMAT(23X, 'ANORM = ' ,E16.8)
   76 FORMAT(1HJ,22X, 'SIZE OF THE READ IN MATRIX =' ,I5)
   77 FORMAT(1HJ,22X, 'FIRST SUBMATRIX GRID FOR CROSS-CORRELATION =',I5)
   78 FORMAT(1HJ,22X, 'SECOND SUBMATRIX FOR CROSS-CORRELATION =',I5)
   79 FORMAT(1HJ,22X, 'SUBMATRIX SIZE FOR THE OBTAINED USEFUL CROSS-CORR
     1ELATIONS =',I5)
   80 FORMAT(1HJ,22X, 'TOTAL MATRIX SIZE OF THE OBTAINED CROSS-CORRELATI
     1ONS =',I5)
  601 FORMAT(1H1,60X,'INPUT DATA'  )
  602 FORMAT(1HT,60X,10(1H*))
      READ(5,10) N4
      READ(5,10) (N(J),J=1,3)
      READ(5,10)  NHALF
      READ(5,10) NTAO
      READ(5,20) DELT
      READ(5,10) ISS
      NOPTS=N(1)*N(2)
      NN=N(1)
      NNN=N(2)
C
C
      JDIME=104
      CALL  ZEROIJ(TEMP,JDIME,JDIME)
      CALL  ZEROIJ(TV,JDIME,JDIME)
      DO 1 I=1,N4
      READ(5,35)  (TV(I,J),J=1,N4)
    1 CONTINUE
C
C     WE WRITE OUT THE VALUE OF ISS, INDICATING WHETHER THE CORRELATIONS
C     TO BE OBTAINED ARE ON THE FILTERED DATA OR ON THE UNFILTERED DATA.
C
      WRITE(6,70)  ISS
C
C     TO CALCULATE AND PRINT OUT THE NYQUIST FREQ. AND OTHER NECESSARY
C     DATA.
```

```
C
      FN=1./(2.*DELT)
      WRITE(6,60) DELT,FN
C
      WRITE(6,46)
      WRITE(6,601)
      WRITE(6,602)
      CALL  SPLIT(N4,N4,TV)
C
C     TO OBTAIN CROSS-CORRELATION AND CROSS-POWER SPECTRA BY
C     TAKING A MATRIX OF SIZE N(1)*N(2) AT A TIME, WE SET
C     UP A LOOP SYSTEM
C
C     NUMBER OF STEPS IN A ROW OR IN A COLUMN.
      ISTEPS=(N4-NN+NHALF)/NHALF
      DO 3 IX=1,ISTEPS
      DO 3 JX=1,ISTEPS
      DO 2 I=1,NN
      DO 2 J=1,NNN
      K1=(IX-1)*NHALF+I
      K2=(JX-1)*NHALF+J
      WORK2(I,J)=TV(K1,K2)
      WORK1(I,J)=0.0
    2 CONTINUE
C     REMOVE D.C. FROM EACH SAMPLE MAP.
      CALL  DC OUT(WORK2,1,NN)
      K1=(NN-NHALF)/2+1
      K2=(NN+NHALF)/2
      DO 4 I=K1,K2
      DO 4 J=K1,K2
      WORK1(I,J)=WORK2(I,J)
    4 CONTINUE
C
C     INITIALLY WE TRANSFER THE DATA IN THE IMAG. VECTORS 'BY' AND 'DY',
C     FROM WHICH WE WILL TRANSFER TO 'AX' AND 'CX' AFTER ADDING THE
C     NECESSARY ZEROS FOR CROSS-CORRELATION.
C
      CALL  ZERO(DY,NOPTS)
      K=1
      DO 104 I=1,NN
      DO 104 J=1,NNN
      DY(K)=WORK1(I,J)
      K=K+1
  104 CONTINUE
C
      CALL  ZERO(BY,NOPTS)
      K=1
      DO 102 I=1,NN
      DO 102 J=1,NNN
      BY(K)=WORK2(I,J)
      K=K+1
  102 CONTINUE
C
```

```
C     NOW WE CALCULATE CROSS-CORRELATION AND POWER SPECTRA
C     AND PRINT OUT  THE RESULTS FOR EACH CASE
C
      CALL  CROSS
C
      NN=NN-NTAO
      NNN=NNN-NTAO
      N(1)=NN
      N(2)=NNN
      NOPTS=N(1)*N(2)
C     WE STORE THE USEFUL CORRELATIONS OF NHALF BY NHALF SIZE FROM
C     VECTOR X TO MATRIX TEMP(I,J). AND FINALLY WE WILL WRITE OUT THE
C     MATRIX TEMP(I,J).
C
      N1HALF=NHALF-1
      DO 11 I=1,N1HALF
      DO 11 J=1,N1HALF
      J1=((NN+NTAO-NHALF+I)*(NN+NTAO)+(NN+NTAO)-NHALF+1+J)
      WORK1(I,J)=X(J1)
      DO 11 K=1,NHALF
      K1=NHALF-1+K
      K2=((NN+NTAO-NHALF+I)*(NN+NTAO)+K)
      WORK1(I,K1)=X(K2)
   11 CONTINUE
      DO 12 I=1,NHALF
      DO 12 J=1,NHALF
      I1=NHALF-1+I
      J1=((I-1)*(NN+NTAO)+J)
      J2=NHALF-1+J
      WORK1(I1,J2)=X(J1)
      DO 12 K=1,N1HALF
      K1=((I-1)*(NN+NTAO)+(NN+NTAO)-NHALF+1+K)
      WORK1(I1,K)=X(K1)
   12 CONTINUE
      N2HALF=(2*NHALF)-1
      DO 5 I=1,N2HALF
      DO 5 J=1,N2HALF
      K1=(IX-1)*N2HALF+I
      K2=(JX-1)*N2HALF+J
      TEMP(K1,K2)=WORK1(I,J)
    5 CONTINUE
    3 CONTINUE
      NT=ISTEPS*N2HALF
C
C     NORMALIZATION OF THE FINAL RESULTS OF CROSS-CORRELATIONS.
C
C     TO OBTAIN THE MAXIMUM ELEMENT FROM THE COMPLETE SET OF
C     CROSS-CORRELATIONS.
C
      ANORM=ABS(TEMP(1,1))
      DO 13 I=1,NT
      DO 13 J=1,NT
      IF(ABS(TEMP(I,J)).GT.ANORM)  GO TO 71
```

```
      GO TO 13
   71 ANORM=ABS(TEMP(I,J))
   13 CONTINUE
      IF(ANORM.EQ.0.0)  GO TO 72
C
C     DIVIDE BY ANORM TO NORMALIZE THE RESULTS.
C     MULTIPLY THE NORMALIZED VALUE BY 1000.0 FOR PLOTING PURPOSE.
C
      DO 14 I=1,NT
      DO 14 J=1,NT
      TEMP(I,J)=1000.0*(TEMP(I,J)/ANORM)
   14 CONTINUE
C
C     END OF NORMALIZATION.
C
   72 WRITE(6,73)  ANORM
C
C     PRINT OUT OF SOME PARAMETERS.
C
      WRITE(6,76)  N4
      WRITE(6,77)  NN
      WRITE(6,78)  NHALF
      WRITE(6,79)  N2HALF
      WRITE(6,80)  NT
C     TO PRINT OUT COMPLETE CROSS-CORRELATION RESULTS.
      WRITE(6,46)
      WRITE(6,7)
      WRITE(6,8)
      CALL  SPLIT(NT,NT,TEMP)
      DO 15 I=1,NT
      WRITE(7,35)  (TEMP(I,J),J=1,NT)
   15 CONTINUE
      STOP
      END
```

```
      SUBROUTINE  CROSS
C
C     THIS SUBROUTINE CALCULATES CROSS-CORRELATION AND POWER SPECTRA
C     AND PRINTS RESULTS
C
C
C     ANY CHANGE OF DIMENSION CARD NEED ALSO A CHANGE IN THE
C     IDIME=2500 CARD.
C
      COMMON AX(2500),BY(2500),CX(2500),DY(2500),S(2500),X(2500),Y(2500)
      COMMON  DELT
      COMMON  N(3)
      COMMON  NN,NNN,NOPTS,NTAO,NHALF
      REAL  AX,BY,CX,DY,S,X,Y,DELT
      INTEGER  N,NN,NNN,NOPTS,NTAO
C
C     FIRST WE SET ALL VALUES OF THE 'AX' AND 'CX'- ARRAY EQUAL TO ZERO
C
      IDIME=2500
      CALL ZERO(AX,IDIME)
      CALL ZERO(CX,IDIME)
C
C     THE MATICES 'BY' AND 'DY' ARE IN THE NORTHWEST QUADRANTS,
C     FROM WHICH WE WISH TO FORM LARGE MATRIX 'AX' AND 'CX' WITH
C     ZEROS IN OTHER QUADRANTS
C
      DO 2 KK=1,NN
      K=KK-1
      M1=K*(NN+NTAO)+1
      M2=K*(NN+NTAO)+NN
      DO 2 J=M1,M2
      M3=J-(K*NTAO)
      AX(J)=BY(M3)
      CX(J)=DY(M3)
    2 CONTINUE
      NN=NN+NTAO
      NNN=NNN+NTAO
      N(1)=NN
      N(2)=NNN
      NOPTS=N(1)*N(2)
C
C     TO OBTAIN FOURIER TRANSFORMS
C
      CALL ZERO(BY,NOPTS)
      CALL ZERO(S,NOPTS)
      CALL ARMDFT(N,AX,BY,S)
      CALL ZERO(DY,NOPTS)
      CALL ZERO(S,NOPTS)
      CALL ARMDFT(N,CX,DY,S)
C
C     WE CALCULATE CROSS-POWER AND STORE REAL PART IN X-VECTOR
C     AND THE IMAG. PART IN Y-VECTOR. THUS WE OBTAIN CROSS POWER =X(J)+I
C
```

```
      J=0
      DO 80 KK=1,NN
      DO 80 JJ=1,NNN
      J=J+1
      X(J)=AX(J)*CX(J)+BY(J)*DY(J)
      Y(J)=BY(J)*CX(J)-AX(J)*DY(J)
   80 CONTINUE
C
C     TO OBTAIN CROSS-CORRELATION WE HAVE TO TAKE THE
C     INVERSE FOURIER TRANSFORMS OF X(J)+IY(J)
C
C     FIRST WE REPLACE FOURIER COEFFS. BY COMPLEX CONJUGATE
C
      CALL CONJU(Y,NOPTS)
      CALL ZERO(S,NOPTS)
      CALL ARMDFT(N,X,Y,S)
C
C     MULTIPLYING FACTOR OF INVERSE FOURIER TRANSFORMS
C
      J=0
      DO 90 J1=1,NOPTS
      J=J+1
      X(J)=X(J)/FLOAT(NOPTS)
      Y(J)=Y(J)/FLOAT(NOPTS)
   90 CONTINUE
C
C     TO COMPLETE THE INVERSE FOURIER TRANSFORM WE SHOULD
C     TAKE THE COMPLEX CONJUGATE AGAIN. ACTUALLY IT IS NOT
C     NECESSARY AS WE ARE INTERESRED IN CROSS-CORRELATION WHICH
C     IS REAL AND FOR THIS, IMAG. PART SHOULD BE ZERO
C
      CALL CONJU(Y,NOPTS)
C
C     TO MULTIPLY WITH THE NORMALIZING CORRELATION FACTOR
C
      J=0
      DO 100 J2=1,NOPTS
      J=J+1
      X(J)=X(J)/FLOAT(NHALF*NHALF)
      Y(J)=Y(J)/FLOAT(NHALF*NHALF)
  100 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE  ZEROIJ(WORK,N,M)
C     IT ZERO'S THE MATRIX WORK(I,J)
      REAL  WORK(104,104)
      DO 1 I=1,N
      DO 1 J=1,M
      WORK(I,J)=0.0
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE ZERO(X,NCPTS)
C     SETS THE VECTOR TO ZERO
      REAL X(10)
      DO 1 J=1,NCPTS
      X(J)=0.0
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CONJU(Y,NOPTS)
C     TAKES THE COMPLEX CONJUGATE OF THE IMAG. PART
      REAL Y(10)
      J=0
      DO 1 J1=1,NOPTS
      J=J+1
      F=Y(J)
      Y(J)=-F
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE  SPLIT(N,M,X)
C
C     THIS PROGRAM SPLITS THE WRITE STATEMENT IN THE FORMAT I=1,N AND
C     J=1,26 AT A TIME.
C
C     WE CAN JOIN THE DIFFERENT WRITTEN PARTS IN THE FORM OF A MAP.
C
      REAL  X(104,104)
   45 FORMAT(1X,26F5.0)
   46 FORMAT(1H1)
   50 FORMAT(1HJ)
      M1=M/26
      IF(M1.LT.1)  GO TO 4
      DO 1 IC=1,M1
      I1=(26*(IC-1))+1
      I2=I1+25
      DO 2 I=1,N
      WRITE(6,45)  (X(I,J),J=I1,I2)
      WRITE(6,50)
      WRITE(6,50)
    2 CONTINUE
      WRITE(6,46)
      WRITE(6,50)
      WRITE(6,50)
    1 CONTINUE
      IF((M1*26).EQ.M)  RETURN
    4 I3=(26*M1)+1
      DO 3 I=1,N
      WRITE(6,45)  (X(I,J),J=I3,M)
      WRITE(6,50)
      WRITE(6,50)
    3 CONTINUE
      WRITE(6,46)
      RETURN
      END
```

APPENDIX E

1. Least squares fit (for a straight line) using first degree polynomial.

By the method of least squares one can assign weights to the various points of the graph and determine the best fitting straight line through them. A linear relation for a straight line can be expressed as:

$$y = a + bx \qquad \text{.....(E-1)}$$

In equation (E-1) 'b' is the slope of the line and 'a' gives its intercept. Suppose we have n by n equally spaced matrix whose coordinates are at $(x_1,y_1),(x_1,y_2),\ldots,(x_1,y_n)$; (x_2,y_1), $(x_2,y_2)\ldots,(x_2,y_n)$;... and (x_n,y_1), (x_n,y_2), $\ldots,(x_n,y_n)$. The weighting factors for these coordinates are denoted by $w(x_1,y_1)$, $w(x_1,y_2),\ldots,w(x_1,y_n)$; $w(x_2,y_1)$, $w(x_2,y_2)\ldots,w(x_2,y_n)$;... and $w(x_n,y_1)$, $w(x_n,y_2),\ldots,w(x_n,y_n)$.

It can be easily shown that

$$a = \frac{\Sigma w_i y_i \Sigma w_i x_i^2 - \Sigma w_i y_i x_i \Sigma w_i x_i}{\Sigma w_i \Sigma w_i x_i^2 - (\Sigma w_i x_i)^2} \qquad \text{.....(E-2)}$$

and

$$b = \frac{\Sigma w_i y_i \Sigma w_i x_i - \Sigma w_i y_i x_i \Sigma w_i}{(\Sigma w_i x_i)^2 - \Sigma w_i x_i^2 \Sigma w_i} \qquad \text{.....(E-3)}$$

2. Equation of a normal

The equation of a normal to the straight line given by equations (E-1), (E-2) and (E-3), is

$$y-y' = -\frac{1}{b}(x-x')$$

or

$$y = y' + \frac{x'}{b} - \frac{x}{b} \qquad \text{.....(E-4)}$$

If x' and y' are the coordinates of the maximum point through which the normal is to be found, then

$$a' = \frac{1}{b}(x'+by') \qquad \text{.....(E-5)}$$

3. Minimum distance between the two points

It is known that the minimum distance from any point $p_1(x_1,y_1)$ to a point $p(x,y)$ on the line

$$ax+by+c = 0 \qquad \text{.....(E-6)}$$

is given by

$$P_1P = \frac{|ax_1+by_1+c|}{\sqrt{(a^2+b^2)}} \qquad \text{.....(E-7)}$$

Thus the distance d from any point (x,y) to a point (x_1,y_1) on the normal line of equation (E-5) is easily obtained. Comparing equations (E-5) and (E-6) we have $a = 1$, $b = b$ and $c =-a'b$, substituting these values in Equation (E-7) we obtain

$$d = \frac{x+by-a'b}{\sqrt{(1+b^2)}} \qquad .....(E-8)$$

From equations (E-5) and (E-8), we obtain

$$d = \frac{(x-x_1)+b(y-y_1)}{\sqrt{(1+b^2)}} \qquad .....(E-9)$$

Where x_1, y_1 represents the coordinates of the last point through which the equation of the normal line passes.

It is possible that in the evaluation of equations (E-3) or (E-9) the denominator may be zero. To avoid such a situation, we may consider equation (E-1) in the following form i.e.

$$y = \frac{a_2}{b_2} + \frac{b_1}{b_2} x \qquad .....(E-10)$$

Where $a = \frac{a_2}{b_2}$ and $b = \frac{b_1}{b_2}$. for $|b_2|<+\infty$ and $|b_1|<+\infty$

Equation of a normal becomes

$$b_2x'+b_1y' - (b_1y+b_2x) = 0 \qquad .....(E-11)$$

and the minimum distance between the two points can be written as

$$d = \frac{b_2x_1+b_1y_1 - (b_1y+b_2x)}{\sqrt{(b_1{}^2+b_2{}^2)}} \qquad .....(E-12)$$

In our calculation's, equation (E-12) has been used.

The sign of d in equation (E-12) will be the same on one side of the normal line and opposite on the other side of the line. We have used this criteria for scanning the maximum points in a particular direction and rejecting in the opposite direction i.e. on the other side of the normal line. The program is designed to compare the distance between the nearest points to the normal line with the distance between the last two maxima point coordinates.

4. Least squares fit using third degree polynomials

Consider the following 3rd degree polynomial equation to fit the trend points as obtained from the cross-correlation matrix.

$$y = B_1+B_2x+B_3x^2+B_4x^3 \qquad \text{.....(E-13)}$$

B_1, B_2, B_3 and B_4 are constants and are determined from the coordinates of the n trend points as obtained by a least square procedure. In other words B_1, B_2, B_3 and B_4 are determined in such a way that

$$\sum_{i=1}^{n} w_i(\delta y_i)^2 = \text{minimum} \qquad \text{.....(E-14)}$$

or

$$\sum_{i=1}^{n} w_i(y_i-B_1-B_2x_i-B_3x_i^2-B_4x_i^3)^2 = \text{minimum} \qquad \text{.....(E-15)}$$

Where w_i are the weighting factors for the various δy_i's.

Differentiating equation (E-15) with respect to B_1, B_2, B_3 and B_4 respectively, we obtain the following equation in a matrix form:

$$\begin{vmatrix} \Sigma w_i x_i^0 & \Sigma w_i x_i & \Sigma w_i x_i^2 & \Sigma w_i x_i^3 \\ \Sigma w_i x_i & \Sigma w_i x_i^2 & \Sigma w_i x_i^3 & \Sigma w_i x_i^4 \\ \Sigma w_i x_i^2 & \Sigma w_i x_i^3 & \Sigma w_i x_i^4 & \Sigma w_i x_i^5 \\ \Sigma w_i x_i^3 & \Sigma w_i x_i^4 & \Sigma w_i x_i^5 & \Sigma w_i x_i^6 \end{vmatrix} \begin{vmatrix} B_1 \\ B_2 \\ B_3 \\ B_4 \end{vmatrix} = \begin{vmatrix} \Sigma w_i y_i x_i^0 \\ \Sigma w_i y_i x_i \\ \Sigma w_i y_i x_i^2 \\ \Sigma w_i y_i x_i^3 \end{vmatrix} \quad \text{.....(E-16)}$$

More briefly, in matrix notation equation (E-16) becomes

$$\vec{C} \cdot \vec{B} = \vec{D}$$

where

$$C_{jk} = \sum_{i=1}^{n} w_i x_i^{(j+k-2)}$$

$$D_\ell = \sum_i w_i y_i x_i^{\ell-1} \quad \text{.....(E-17)}$$

$$\vec{B} = [B_1, B_2, B_3, B_4]$$

It is to be noted that the matrix $\vec{C}$ is symmetric and to evaluate it one needs to perform only 7 summations [i.e. $S_p = \Sigma w_i x_i^{p-1}$ where $p = 1,2,\ldots,7$]. Thus using any program for solving the set of simultaneous equations, it is easy to obtain the coefficients B_1, B_2, B_3 and B_4 from equation (E-17). Knowing these constants, y values can be calculated for various x values using equation (E-13).

These values of x and y provide the required 3rd degree polynomial least square fit through these points. In the present study, equation's (E-17) were solved by using the Wilkinson's method for the solution of ill-conditioned linear simultaneous equations (Subroutine CS009A by K.F.MAY, Dept. of Computing Science, University of Alberta).

5. <u>Transformation of origin</u>

In equation (E-13), x and y have the origin in the NW corner as shown in figure (E-1), with x-positive eastward and y-positive downward.

For convenience, we may transform the origin to the centre of the matrix as indicated in figure (E-2).

By substituting

$$x = \bar{x} + t$$

and $\qquad$ where $t = \left(\frac{N+1}{2}\right)$

$$y = -\bar{y} + t$$

for N as odd

we obtain the following equation.

$$\bar{y} = -B_1-(B_2-1)t-B_3t^2-B_4t^3-(B_2+2B_3t+3B_4t^2)\bar{x}-(B_3+3B_4t)\bar{x}^2-B_4\bar{x}^3 \qquad \text{.....(E-18)}$$

Thus equation (E-13) is replaced with equation (E-18) for 3rd degree polynomial fit with (0,0) at the centre of the odd matrix, with conventional signs.

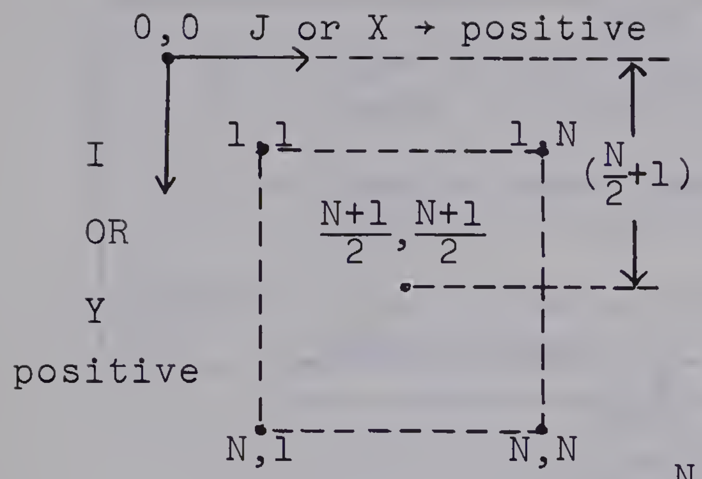


Figure E-1

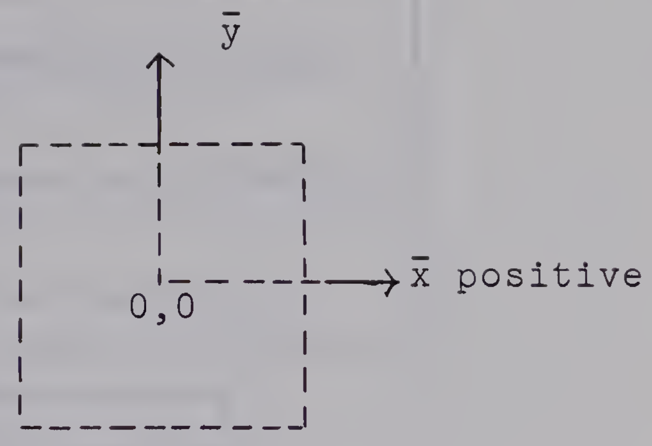


Figure E-2

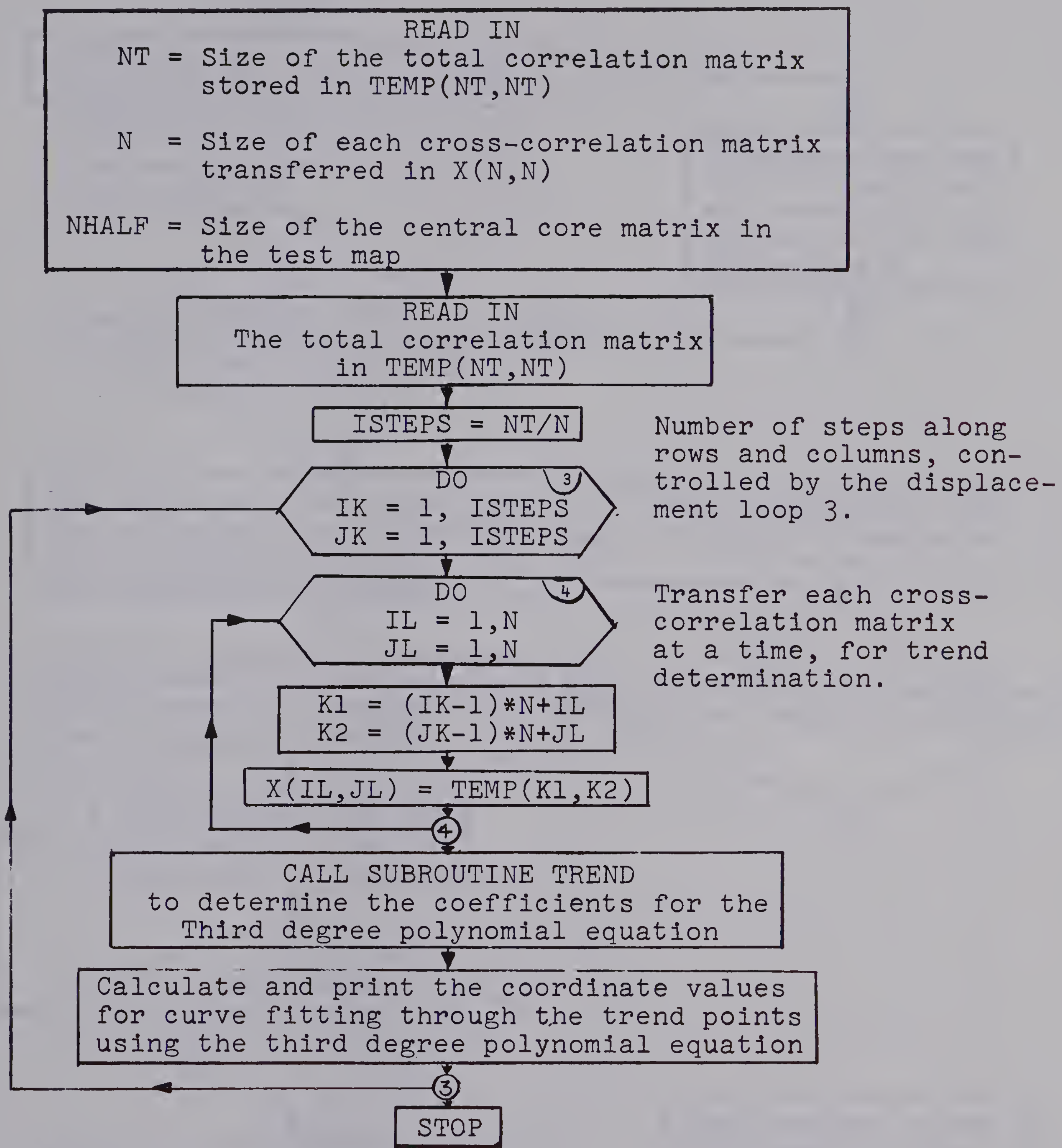
Transformation of origin of the

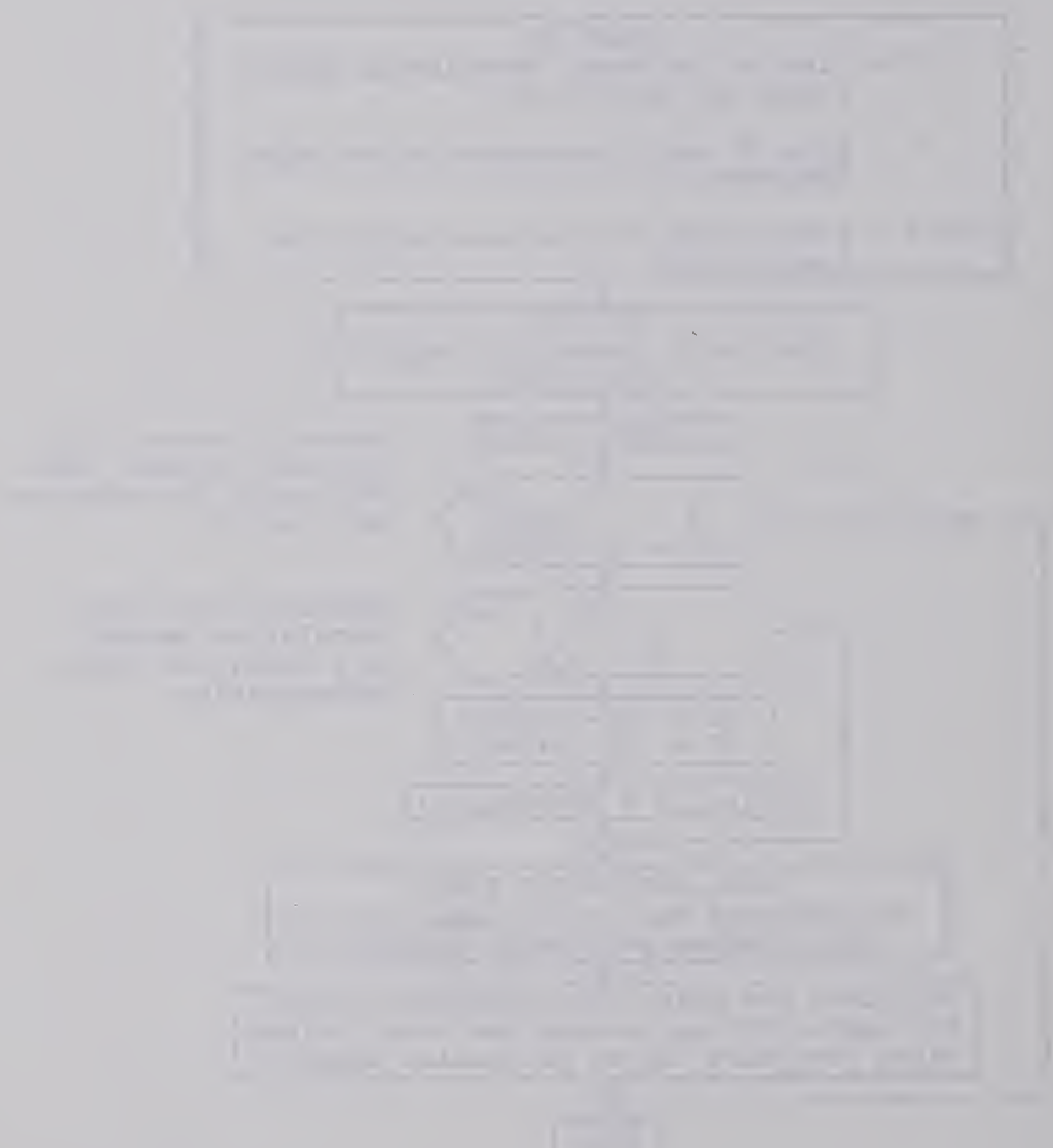
coordinate system

APPENDIX F

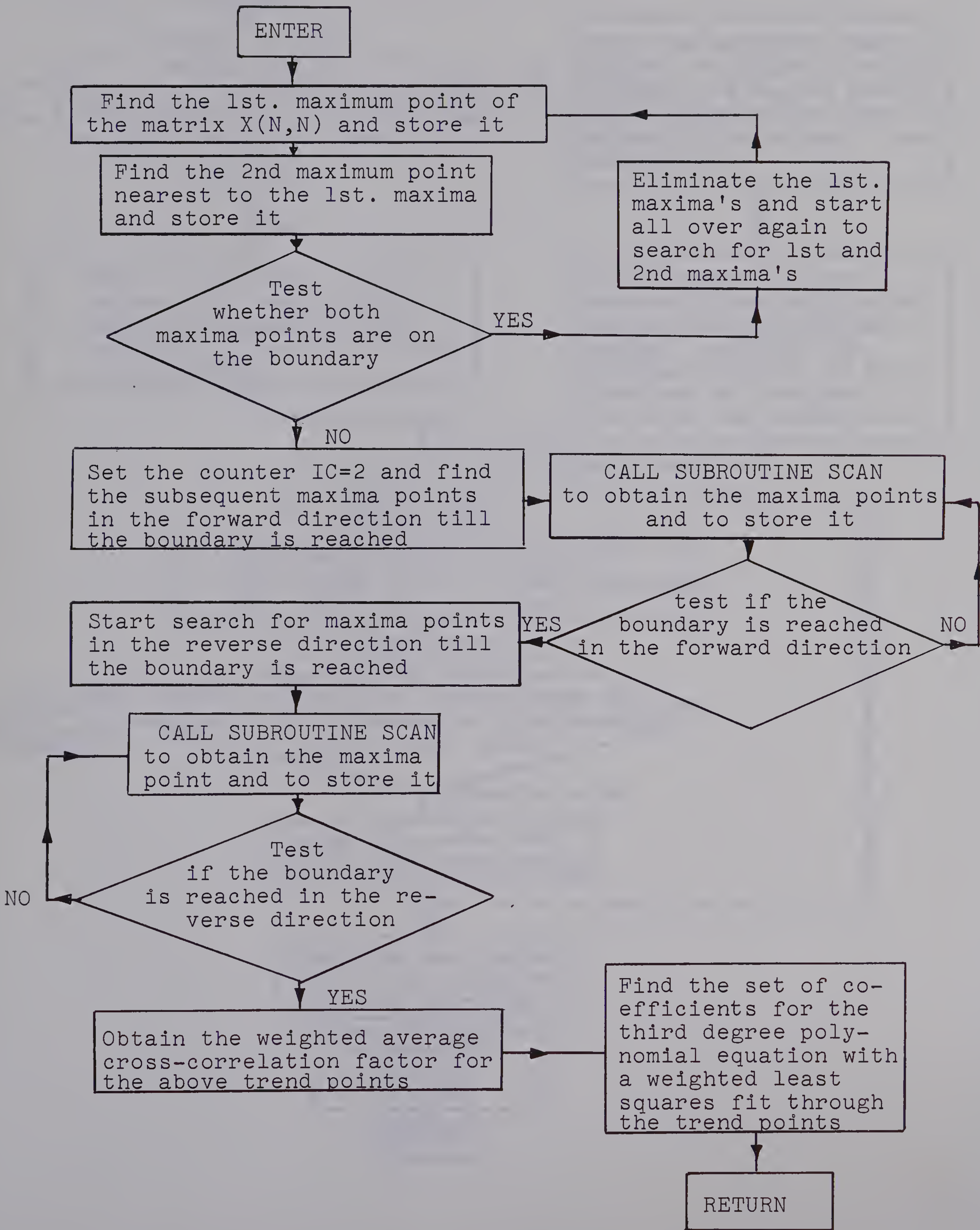
Figure (F-1): Flow diagram for automatic trend analysis.

Main Program

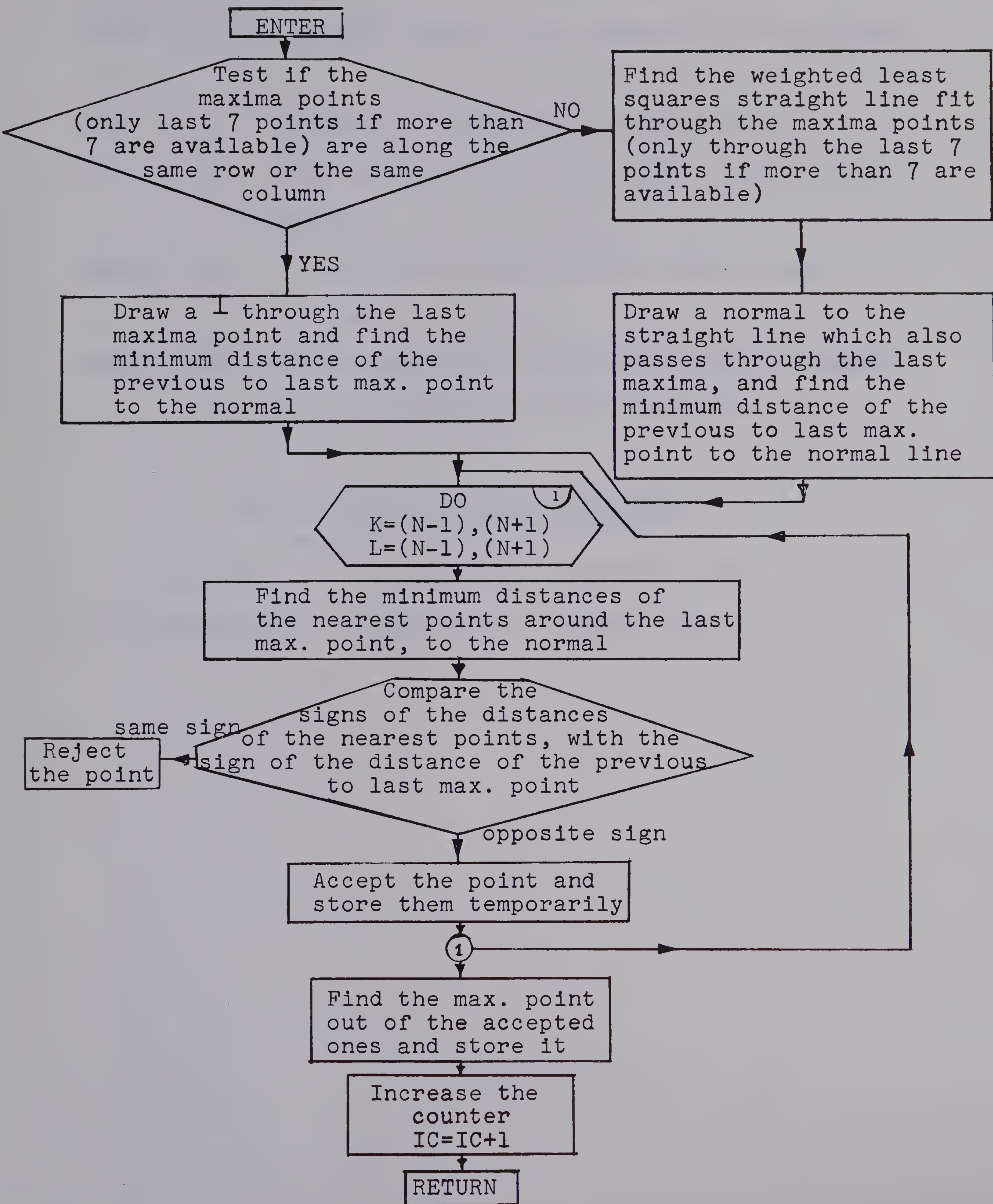




SUBROUTINE TREND



SUBROUTINE SCAN



FORTRAN IV

TREND ANALYSIS PROGRAM USING CROSS-CORRELATION COEFFICIENTS.

NT .... Size of the complete correlation data

N .... Size of each cross-correlation coefficients matrix

NHALF .... is the size of the central core of the test map

TEMP(NT,NT)...is the dimensioned matrix to read in complete set of correlations for the entire data

READ IN THE FOLLOWING PARAMETERS

1. NT,N,NHALF according to FORMAT(5X,3I5)
2. TEMP(N4,N4) according to FORMAT(1X,6E11.4)

```
C      THIS PROGRAM OBTAINS THE MAIN TRENDS FROM EACH SET OF
C      SUB-CORRELATION MATRIX
C
C      IT PRINTS OUT THE 'FIRST AND SECOND MAX. POINT',COORDINATES OF ALL
C      THE MAX. POINTS ON WHICH THIRD DEGREE POLYNOMIAL EQU. IS FITTED,
C      AND PRINTS OUT THE MATRIX EQU.  C(4,4)*B(4)=D(4)
C      WHERE B(4) CONTAINS THE SOLUTION OF 4 UNKNOWN
C      CONSTANTS IN THE THIRD DEGREE POLYNOMIAL EQU.  Y=B1+B2*X+B3*(X**2)
C      +B4*(X**3)
C      IT ALSO PRINTS OUT EACH SUB-CORRELATION MATRIX
C      AND THE AABSCISSEA AND ORDINATE VALUE OF THE FITTED CURVE.
C      NW CORNER IS CONSIDERED AS (1,1) WITH ABSCISSA    EASTWARD
C      AND ORDINATE    DOWNWARD AS POSITIVES.
C
C      NT     SIZE OF THE TOTAL CROSS-CORRELATION MATRIX IN TEMP(NT,NT)
C      N      SIZE OF THE SUB-CROSS-CORRELATION MATRIX IN X(N,N)
C      NHALF SIZE OF THE SMALL TEST MATRIX IN THE ORIGINAL DATA
C            FROM WHICH EACH SUB-CROSS-CORRELATIONS ARE
C            OBTAINED.
       DIMENSION TEMP(104,104),Y(200)
       COMMON   X(25,25),N
       REAL*4   B(4)
       INTEGER  N
C      ANY CHANGE IN COMMON CARD MUST ALSO BE
C      REPLACED IN SUBROUTINES 'TREND' AND 'SCAN'
C
       READ(5,20) NT,N,NHALF
   20  FORMAT(5X,3I5)
       DO 1 K=1,NT
       READ(5,35)(TEMP(K,L),L=1,NT)
   1   CONTINUE
   35  FORMAT(1X,6E11.4)
       ISTEPS=NT/N
       DO 2 IK=1,ISTEPS
       DO 3 JK=1,ISTEPS
       DO 4 IL=1,N
       DO 5 JL=1,N
       K1=(IK-1)*N+IL
       K2=(JK-1)*N+JL
       X(IL,JL)=TEMP(K1,K2)
   5   CONTINUE
   4   CONTINUE
       BABY=0.0
       CALL TREND (B,BABY)
       CALL SPACE (2)
       WRITE(6,9)
   9   FORMAT(1HJ,42X,'INPUT SUB CROSS-CORRELATION MATRIX')
       WRITE(6,10)
   10  FORMAT(1HT,42X,34(1H*))
       DO 11 K=1,N
       WRITE(6,25)(X(K,L),L=1,N)
   11  CONTINUE
   25  FORMAT(1HJ,30X,16F6.0)
```

```
      CALL SPACE(2)
      IF(BABY.EQ.1.0)  GO TO 3
      WRITE(6,16)
      WRITE(6,17)
      WRITE(6,18)
      WRITE(6,19)
      CALL SPACE(1)
   16 FORMAT(1HJ,36X,'THE ORIGIN (0,0) OF COORDINATES X AND Y ')
   17 FORMAT(1HJ,41X,'IS AT THE CENTRE OF THE INPUT MATRIX')
   18 FORMAT(1HJ,36X,'X IS ABSCISSA----POSITIVE EASTWARD FROM ORIGIN')
   19 FORMAT(1HJ,36X,'Y IS ORDINATE----POSITIVE NORTHWARD FROM ORIGIN')
      WRITE(6,12)
   12 FORMAT(1HJ,40X,'ABSCISSA  VALUE',10X,'ORDINATE VALUE')
      WRITE(6,13)
   13 FORMAT(1HT,40X,15(1H*),10X,14(1H*))
      N1=N*10
      N2=N/2+1
      DO 14 K=10,N1
C     SETS THE LIMITS FROM -N/2 TO +N/2
      K1=(2*K-N1-10)/2
      Y(K)=-B(1)-(B(2)-1)*N2-B(3)*(N2**2)-B(4)*(N2**3)-(B(2)+2*B(3)*N2+3
     1*B(4)*(N2**2))*(0.1*K1)-(B(3)+3*B(4)*N2)*((0.1*K1)**2)-B(4)*((0.1*
     2K1)**3)
      IF(Y(K).LT.(-N/2).OR.Y(K).GT.(N/2))  GO TO 14
      XK=FLOAT(K1)/10
      WRITE(6,15)  XK,Y(K)
   14 CONTINUE
   15 FORMAT(45X,F5.1,20X,F5.1)
    3 CONTINUE
    2 CONTINUE
      STOP
      END
```

```
      SUBROUTINE    TREND (B,BABY)
C     AUTOMATIC  TREND  ANALYSIS  PROGRAM
C     IT PRINTS OUT THE  VALUES OF 4 UNKNOWN  CONSTANTS FOR
C     THIRD DEGREE POLYNOMIAL FIT
C
      COMMON   X(25,25),N
      REAL*4  C(4,4),D(4),B(4),A(4,4),RR(4)
      DIMENSION I(100),J(100),II(100),JJ(100),SUM(10)
      INTEGER*4  PIVOT(4)
      INTEGER  N
C     X(N,N)  MATRIX OF DIMENSIONS  N*N
C     N   SIZE OF THE MATRIX X(N,N)
C
C     TO FIND THE FIRST MAXIMUM POINT.  WE STORE IT IN I(1) AND J(1)
      IT=0
      AMAX1=X(1,1)
   20 CONTINUE
      IT=IT+1
      DO 1 K=1,N
      DO 1 L=1,N
      IF(IT.EQ.1)  GO TO 22
C     ELIMINATE THE MAXIMUM POINTS WHICH ARE ON THE BOUNDARY.
      IT1=IT-2
      DO 21 IL=1,IT1,2
      IF(K.EQ.I(IL).AND.L.EQ.J(IL))  GO TO 1
   21 CONTINUE
   22 CONTINUE
      IF(X(K,L).LT.AMAX1)  GO TO 1
      AMAX1=X(K,L)
      I(IT)=K
      J(IT)=L
    1 CONTINUE
C
C     TO FIND THE SECOND MAXIMUM POINT AND STORE IT IN
C     I(2),J(2)
C
C
C     TESTS TO AVOID ZERO INDEXING.
C
      K1=I(IT)-1
      K2=I(IT)+1
      L1=J(IT)-1
      L2=J(IT)+1
      IF(K1.EQ.0)   K1=I(IT)
      IF(L1.EQ.0)   L1=J(IT)
      IT=IT+1
      AMAX2=-1100.0
      DO 2 K=K1,K2
      DO 2 L=L1,L2
C     BYPASS THE CENTRE POINT
      IF(K.EQ.I(IT-1).AND.L.EQ.J(IT-1))  GO TO 2
C     ELIMINATE THE MAXIMUM POINTS WHICH ARE ON THE BOUNDARY.
      IF(IT.EQ.2)  GO TO 24
```

```
      IT1=IT-2
      DO 23 IL=1,IT1,2
      IF(K.EQ.I(IL).AND.L.EQ.J(IL))  GO TO 2
   23 CONTINUE
   24 CONTINUE
C
C     OTHERWISE, WE TEST FOR BOUNDARY CONDITIONS.  IF WE ARE
C     OUTSIDE THE BOUNDARIES OF THE MATRIX, WE BYPASS, OTHERWISE
C     WE CONTINUE
      IF(K.LT.1.OR.L.LT.1) GO TO 2
      IF(K.GT.N.OR.L.GT.N) GO TO 2
C     END OF BOUNDARY CONDITION TEST
C
      IF(X(K,L).LT.AMAX2) GO TO 2
      AMAX2=X(K,L)
      I(IT)=K
      J(IT)=L
    2 CONTINUE
      AMAX1=AMAX2
      IF(I(IT-1).EQ.1.AND.I(IT).EQ.1)  GO TO 20
      IF(J(IT-1).EQ.1.AND.J(IT).EQ.1)  GO TO 20
      IF(I(IT-1).EQ.N.AND.I(IT).EQ.N)  GO TO 20
      IF(J(IT-1).EQ.N.AND.J(IT).EQ.N)  GO TO 20
      IF(I(IT-1).EQ.1.AND.J(IT).EQ.N)  GO TO 20
      IF(J(IT-1).EQ.N.AND.I(IT).EQ.1)  GO TO 20
      IF(J(IT-1).EQ.1.AND.I(IT).EQ.N)  GO TO 20
      IF(I(IT-1).EQ.N.AND.J(IT).EQ.1)  GO TO 20
      IF(I(IT-1).EQ.N.AND.J(IT).EQ.N)  GO TO 20
      IF(J(IT-1).EQ.1.AND.I(IT).EQ.1)  GO TO 20
      IF(I(IT-1).EQ.1.AND.J(IT).EQ.1)  GO TO 20
      IF(J(IT-1).EQ.N.AND.I(IT).EQ.N)  GO TO 20
C     OTHERWISE SET FIRST AND SECOND INDEX OF I,J AS
      I(1)=I(IT-1)
      I(2)=I(IT)
      J(1)=J(IT-1)
      J(2)=J(IT)
C
C     WE START OUR SEARCH FOR SUBSEQUENT MAXIMUM ELEMENTS
C     IN THE FORWARD DIRECTION BY SETTING FLAG=0.0
C     START THE COUNTER AT IC=2
C
      FLAG=0.0
      IC=2
    8 CONTINUE
C     TEST FOR SEARCH IN THE REVERSE DIRECTION, THAT IS, WHEN FLAG=1.0
      IF(FLAG.EQ.1.0) GO TO 9
C     OTHERWISE CALL THE SUBROUTINE SCAN TO FIT THE ST.LINE EQU.
C     AND TO OBTAIN THE NEXT MAXIMUM POINT.
C
      CALL SCAN (IC,I,J,FLAG)
      GO TO 8
    9 CONTINUE
C
```

```
      WRITE(6,100)
  100 FORMAT(1H1,8X,'FIRST MAX. ELEMENT IN',10X,'ROW COORDINATE I(1)',5X
     1,'AND',5X,'COLUMN COORDINATE  J(1)')
C
      WRITE(6,101)
  101 FORMAT(1HT,39X,19(1H*),13X,23(1H*))
C
      WRITE(6,102) I(1),J(1)
  102 FORMAT(46X,I5,28X,I5)
C
      WRITE(6,103)
 103  FORMAT(1HJ,8X,'FORWARD MAX ELEMENTS IN',8X,'ROW COORDINATE I(2)',5
     1X,'AND',5X,'COLUMN COORDINATE J(2)')
C
      WRITE(6,101)
      DO 108 K=1,IC
      WRITE(6,102) I(K),J(K)
 108  CONTINUE
C
C     FOR SEARCH IN THE REVERSE DIRECTION
C     FIRST OF ALL, WE TRANSFER VECTORS I AND J INTO II AND JJ FOR
C     CONVENIENCE THEN CONTINUE THE SEARCH IN THE SAME MANNER
C     AS IN THE FORWARD DIRECTION
C
      DO 10 K=1,IC
      II(K)=I(IC+1-K)
      JJ(K)=J(IC+1-K)
  10  CONTINUE
C
C     TEST WHETHER THE REVERSE DIRECTION IS FINISHED, THAT IS, WHEN
C     FLAG>1.0.  THEN WE CAN PROCEED FOR THIRD DEGREE POLYNOMIAL
C     FIT TO OUR ALL THE MAX. ELEMENTS IN II AND JJ.
  11  CONTINUE
      IF(FLAG.GT.1.0)GO TO 12
      CALL SCAN(IC,II,JJ,FLAG)
      GO TO 11
  12  CONTINUE
C
      WRITE(6,105)
  105 FORMAT(1HJ,8X,'ALL THE MAX ELEMENTS IN',9X,'  ROW COORDINATES  ',4
     1X,'AND',5X,'  COLUMN COORDINATES  ')
      WRITE(6,101)
      DO 104 K=1,IC
      WRITE(6,102) II(K),JJ(K)
  104 CONTINUE
C     TO OBTAIN AVERAGE CROSS-CORRELATION FACTOR
      SUM1=0.0
      DO 25 K=1,IC
      II1=II(K)
      JJ1=JJ(K)
      SUM1=SUM1+X(II1,JJ1)
   25 CONTINUE
      SUM1=SUM1/(IC*1000)
```

```
      CALL  SPACE(2)
      WRITE(6,26)  SUM1
   26 FORMAT(1HJ,8X,'AVERAGE CORRELATION FACTOR =',F10.5)
C
C     TO FIT THE THIRD DEGREE POLYNOMIAL
C     MATRIX OF SIZE 4X4 IS SYMMETRICAL, AND WE NEED TO OBTAIN
C     ONLY 7 SUMMATION SETS.
C
      DO 13 K=1,7
      SUM(K)=0.0
      DO 14 L=1,IC
      II1=II(L)
      JJ1=JJ(L)
      SUM(K)=SUM(K)+X(II1,JJ1)*(JJ1**(K-1))
   14 CONTINUE
   13 CONTINUE
C
C     NOW SET THE COMPLETE MATRIX C(4,4) WITH ITS PROPER
C     SYMMETRIES.
C
      DO 15 K=1,4
      DO 16 L=1,4
      C(K,L)=SUM(K+L-1)
   16 CONTINUE
   15 CONTINUE
C
C     NOW CALCULATING THE SUMMATIONS FOR VECTOR D
C
      DO 17 K=1,4
      SUM(K)=0.0
      DO 18 L=1,IC
      II1=II(L)
      JJ1=JJ(L)
      SUM(K)=SUM(K)+X(II1,JJ1)*II1*(JJ1**(K-1))
   18 CONTINUE
      D(K)=SUM(K)
   17 CONTINUE
C
C     NOW WE HAVE THE MATRIX C AND VECTOR D, THE SOLUTION OF
C     4 UNKNOWN CONSTANTS COULD BE OBTAINED IN VECTOR B
C     BY USING THE PACKAGE SUBROUTINE CS009A, IN IBM 67/360
C     U OF A, FOR SOLVING THE SIMULTANEOUS EQUS.
C
      CALL CS009 A (C,D,B,4,A,RR,PIVOT,BABY)
      CALL SPACE(2)
      IF(BABY.EQ.1.0)  GO TO 30
      WRITE(6,106)
  106 FORMAT(1HJ,31X,' C(K,L) ',43X,' B(K) ',24X,' D(K) ')
      WRITE(6,107)
  107 FORMAT(1HT,31X,8(1H*),43X,6(1H*),24X,6(1H*))
      WRITE(6,19) (((C(K,L),L=1,4),B(K),D(K)),K=1,4)
   19 FORMAT((1HJ,7X,4E14.7,15X,E14.7,15X,E14.7//))
   30 CONTINUE
```

```
RETURN
END
```

```
      SUBROUTINE SCAN(IC,I,J,FLAG)
C     IT IS USED IN SUBROUTINE TREND
C     THIS SUBROUTINE OBTAINS STRAIGHT LINE FIT UP TO 5 POINTS
C     AND FINDS SUBSEQUENT MAXIMUM ELEMENTS.
C
      DIMENSION I(100),J(100),ID(8),IE(8)
      COMMON   X(25,25),N
      INTEGER N
C     TEST FOR SPECIAL CASES
C
C     TEST WHETHER I OR J VECTORS CONTINUE IN THE SAME ROW OR COLUMN
C
C     CASE OF THE SAME ROW
      ISP=0
      JSP=0
      DO 8 K=1,IC
      KK=IC+1-K
      IF(K.GT.7)  GO TO 9
      IF(I(IC).NE.I(KK)) GO TO 10
    8 CONTINUE
    9 CONTINUE
      DIC1=J(IC)-J(IC-1)
C     SET ISP=1, TO CALCULATE DIN FOR THIS CASE
      ISP=ISP+1
C     TEST FOR BOUNDARY CONDITIONS
      IF(I(IC).LE.1.OR.J(IC).LE.1) GO TO  23
      IF(I(IC).GE.N.OR.J(IC).GE.N) GO TO 23
C     END OF B.C. TEST
      GO TO 1
   10 CONTINUE
C
C     CASE OF THE SAME COLUME
C
      DO 11 K=1,IC
      KK=IC+1-K
      IF(K.GT.7)  GO TO 12
      IF(J(IC).NE.J(KK)) GO TO 13
   11 CONTINUE
   12 CONTINUE
      DIC1=I(IC)-I(IC-1)
C     SET JSP=1 TO CALCULATE DIN ACCORDINGLY.
      JSP=JSP+1
C     TEST FOR BOUNDARY CONDITIONS
      IF(I(IC).LE.1.OR.J(IC).LE.1) GO TO 23
      IF(I(IC).GE.N.OR.J(IC).GE.N) GO TO 23
C     END OF B.C. TEST
      GO TO 1
   13 CONTINUE
C     TEST FOR BOUNDARY CONDITIONS
C
      IF(I(IC).LE.1.OR.J(IC).LE.1)  GO TO 23
      IF(I(IC).GE.N.OR.J(IC).GE.N)  GO TO 23
C     END OF B.C. TEST
```

```
C     TO FIND THE PARAMETER B=B1/B2=NUM/DEN FOR THE ST. LINE  Y=A+BX
      SUM1=0.0
      SUM2=0.0
      SUM3=0.0
      SUM4=0.0
      SUM5=0.0
      DO 3 K=1,IC
      KK=IC+1-K
      I1=I(KK)
      J1=J(KK)
C     TEST FOR 7 OR LESS THAN 7 POINTS TO FIT A ST. LINE EQU.
      IF(K.GT.7) GO TO 4
      SUM1=SUM1+X(I1,J1)*J1
      SUM2=SUM2+X(I1,J1)*I1
      SUM3=SUM3+X(I1,J1)*J1*J1
      SUM4=SUM4+X(I1,J1)*J1*I1
      SUM5=SUM5+X(I1,J1)
    3 CONTINUE
    4 CONTINUE
      B1=(SUM2*SUM1-SUM4*SUM5)
      B2=(SUM1*SUM1-SUM3*SUM5)
C
C     SCANNING PROCEEDURE (OUT OF 8 POINTS AROUND THE LAST MAX. POINT)
C     TO FIND THE SUBSEQUENT MAXIMUMS.
C
C     DIC1     DISTANCE CALCULATED BETWEEN THE NORMAL LINE AND PREVIOUS
C              TO LAST MAX. POINT THAT IS (IC-1) POINT.
C
C     DIC2     DISTANCE CALCULATED BETWEEN THE NORMAL LINE AND PREVIOUS
C              TO LAST TO LAST MAX. POINT THAT IS (IC-2) POINT.
C
C     DIN      DISTANCE OF VARIOUS (8 POINTS) POINTS TO BE COMPARED WITH
C              DIC1, TO CHECK WHETHER TO ACCEPT THE POINT OR NOT WHILE
C              PROCEEDING IN A PARTICULAR DIRECTION.
C     THE ACCEPTED POINTS ARE TEMPORARILY STORED IN VECTORS ID AND IE.
C
      DIC1=(B2*J(IC-1)+B1*I(IC-1)-B1*I(IC)-B2*J(IC))/SQRT(B1*B1+B2*B2)
      IF(IC.EQ.2)  GO TO 1
      DIC2=(B2*J(IC-2)+B1*I(IC-2)-B1*I(IC)-B2*J(IC))/SQRT(B1*B1+B2*B2)
    1 CONTINUE
C
C     INITIALLY SETTING THE VECTORS ID AND IE TO ZERO.
      DO 20 K=1,8
      ID(K)=0
      IE(K)=0
   20 CONTINUE
C
C     TESTS TO AVOID ZERO INDEXING.
C
      K1=I(IC)-1
      K2=I(IC)+1
      L1=J(IC)-1
      L2=J(IC)+1
```

```
      IF(K1.EQ.0)   K1=I(IC)
      IF(L1.EQ.0)   L1=J(IC)
      ISC=0
      DO 6 K=K1,K2
      DO 5 L=L1,L2
C     BYPASS ALL THE PREVIOUS MAX. POINTS.
      DO 33 IK=1,IC
      IF(K.EQ.I(IK).AND.L.EQ.J(IK))  GO TO 5
   33 CONTINUE
C     OTHERWISE TEST FOR BOUNDARY CONDITIONS.
      IF(K.LT.1.OR.L.LT.1)  GO TO 5
      IF(K.GT.N.OR.L.GT.N)  GO TO 5
C     END OF B.C. TEST.
C     NOW OBTAIN DISTANCE DIN FOR EACH OF 8 POINTS TO THE
C     NORMAL LINE AND COMPARE WITH DIC1.
C     DIN WOULD BE NEGATIVE OF DIC1 ON THE OPPOSITE SIDE OF THE
C     NORMAL LINE AND POSITIVE ON THE SAME SIDE OF NORMAL LINE.
      IF(ISP.EQ.1) GO TO 14
      IF(JSP.EQ.1) GO TO 15
C
      DIN=(B2*L+B1*K-B1*I(IC)-B2*J(IC))/SQRT(B1*B1+B2*B2)
C     TAKING INTO ACCOUNT THE COMPUTATIONAL ERROR LIMITS.
      DIN1=DIN+0.00001
      DIN2=DIN-0.00001
      GO TO 19
C
   14 CONTINUE
      DIN=J(IC)-L
C     TAKING INTO ACCOUNT THE COMPUTATIONAL ERROR LIMITS.
      DIN1=DIN+0.00001
      DIN2=DIN-0.00001
      GO TO 16
   15 CONTINUE
      DIN=I(IC)-K
C     TAKING INTO ACCOUNT THE COMPUTATIONAL ERROR LIMITS.
      DIN1=DIN+0.00001
      DIN2=DIN-0.00001
      GO TO 16
   19 CONTINUE
C     TEST WHETHER WE ARE IN THE FORWARD DIRECTION OR IN THE
C     REVERSE DIRECTION.
C     THEN COMPARE DIN AND DIC1 ACCORDINGLY.
C
C     CHECK THE SIGN OF DIC1 WITH RESPECT TO DIC2 AND THEN
C     COMPARE DIN AND DIC1 ACCORDINGLY.
C
      IF(IC.EQ.2)  GO TO 16
      IF(DIC1.LT.(0.0).AND.DIC2.LT.(0.0))  GO TO 16
      IF(DIC1.GT.(0.0).AND.DIC2.GT.(0.0))  GO TO 16
      GO TO 17
C
   16 CONTINUE
      IF(DIN1.LT.(0.0).AND.DIC1.LT.(0.0))  GO TO 5
```

```
      IF(DIN2.GT.(0.0).AND.DIC1.GT.(0.0))  GO TO 5
      GO TO 18
   17 CONTINUE
      IF(DIN2.LT.(0.0).AND.DIC1.GT.(0.0))  GO TO 5
      IF(DIN1.GT.(0.0).AND.DIC1.LT.(0.0))  GO TO 5
   18 CONTINUE
C     OTHERWISE ACCEPT THIS VALUE AND STORE TEMPORARILY IN VECTOR
C     ID AND IE.
C     INCREASE THE COUNTER
      ISC=ISC+1
      ID(ISC)=K
      IE(ISC)=L
    5 CONTINUE
    6 CONTINUE
C
      I1=ID(1)
      J1=IE(1)
      IF(I1.EQ.0.OR.J1.EQ.0)  GO TO 23
      I3=IC+1
      AMAX3=X(I1,J1)
      DO 7 K=1,ISC
      I2=ID(K)
      J2=IE(K)
      IF(X(I2,J2).LT.AMAX3) GO TO 7
      AMAX3=X(I2,J2)
      I(I3)=I2
      J(I3)=J2
    7 CONTINUE
C
C     INCREASE THE COUNTER
      IC=IC+1
C     CHECK WHETHER X(I(IC),(J(IC))  IS LESS THAN OR EQUAL TO 0.0,
C     IF IT IS ZERO WE QUIT IN THIS DIRECTION.
      IF(X(I(IC),J(IC)).LE.0.0)  GO TO 23
      GO TO 21
   23 CONTINUE
C     INCREASE FLAG FOR REVERSE DIRECTION OR FOR
C     COMPLETION OF SCANNING AND RETURN
      FLAG=FLAG+1.0
   21 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SPACE (N)
C     OUTPUT SPACER
C
      INTEGER N
C
      INTEGER K
C
      IF (N.GE.60) WRITE (6,110)
      IF (N.GE.60) RETURN
      K=N
 100  IF (K.LE.0) RETURN
      K=K-1
      WRITE (6,120)
      GO TO 100
 110  FORMAT (1H1)
 120  FORMAT (1H )
      RETURN
      END
```

```
      SUBROUTINE CSC09A(AA,RHS,XX,NN,BB,RR,PIVCT,BABY)
C
C     ..............WILKINSCN'S ALGORITHM
C
C     ..............USAGE
C                   -----
C                   CALL CSC09A(AA,RHS,XX,NN,BB,RR,PIVOT,BABY)
C                             AND
C                   CALL CSC09B(RHS,XX)
C
C     ..............ENTRY CSC09A
C                   ------------
C     ..............SOLUTICN OF (ILL-CONDITIONED) SYSTEM AA*XX=RHS OF
C     ..............SIMULTANEOUS LINEAR EQUATIONS TO (NEAR) FULL
C     ..............WORKING SINGLE PRECISION ACCURACY FOR THE
C     ..............FIRST RIGHT HAND SIDE-RHS.
C                   AA........THE MATRIX OF THE SYSTEM
C                   RHS.......THE INITIAL RIGHT HAND SIDE
C                   XX........SOLUTION ASSOCIATED WITH RHS
C                   NN........ORDER OF SYSTEM
C                   BB........LU FACTORIZATION OF AA
C                   RR........CORRECTION VECTOR
C                   PIVOT.....WORKSPACE VECTOR
C
C     ..............ENTRY CSC09B
C                   ------------
C     ..............SOLUTICN OF SYSTEM FOR SUBSEQUENT RHS.
C      .............IT CAN ALSO BE USED REPEATEDLY TO OBTAIN
C     ..............THE INVERSE MATRIX.
C                   RHS.......A RIGHT HAND SIDE FOR THE SYSTEM
C                   XX........SOLUTION ASSOCIATED WITH THE RHS
C
      REAL*8 SS,RCUND
      REAL*4 AA(NN,NN),BB(NN,NN),       XX(NN),RR(NN),NORMEE,NORMXX
     1,RHS(NN)
      INTEGER*4 NN,R,RBIG,PIVOT(NN),FIN
      N=NN
      RCUND=DBLE(2.0**(-25))
C     ..............OBTAIN A WORKSPACE COPY BB OF MATRIX AA
      DC 1 I=1,N
      DO 1 J=1,N
    1 BB(I,J)=AA(I,J)
C     ..............DETERMINATION OF THE MAXIMUM ROW ELEMENTS
      DO 3 I=1,N
      RR(I)=0.0
      DO 2 J=1,N
    2 RR(I)=AMAX1(RR(I),ABS(AA(I,J)))
C     ..............2**-25 IS USED AS AN EFFECTIVE ZERO
      IF(RR(I).LT.2.98E-08) GO TO 1000
    3 CONTINUE
C     ..............L*U FACTORIZATION OF MATRIX BB
      R=1
    4 BIG=0.0
```

```
      RBIG=R
      II=R
C     ..............(D.P.) ACCUMULATION OF LL&UU PRODUCTS
C     ..............DETERMINATION OF LL ELEMENTS
   41 JJ=1
      SP=BB(II,R)
    5 CONTINUE
      IF(JJ.EQ.R) GO TO 6
      SP=SP-BB(II,JJ)*BB(JJ,R)
      JJ=JJ+1
      GO TO 5
    6 BB(II,R)=SP
      HOLD=ABS(BB(II,R)/RR(II))
      IF(HOLD-BIG) 7,7,8
    8 BIG=HOLD
      RBIG=II
    7 IF(II.EQ.N) GO TO 9
      II=II+1
      GO TO 41
    9 IF(BIG.LT.2.98E-08) GO TO 2000
      RR(RBIG)=RR(R)
      PIVOT(R)=RBIG
C     ..............INTERCHANGE ROWS R AND PIVOT(R)
      DO 10 I=1,N
      HOLD=BB(R,I)
      BB(R,I)=BB(RBIG,I)
   10 BB(RBIG,I)=HOLD
C     ..............DETERMINATION OF UU ELEMENTS
      II=R
  101 IF(II.NE.N) GO TO 11
      IF(R.EQ.N) GO TO 66
      R=R+1
      GO TO 4
   11 II=II+1
      JJ=1
      SP=BB(R,II)
   12 CONTINUE
      IF(JJ.EQ.R) GO TO 13
      SP=SP-BB(R,JJ)*BB(JJ,II)
      JJ=JJ+1
      GO TO 12
   13 BB(R,II)=SP/BB(R,R)
      GO TO 101
C     ..............FACTORIZATION OF AA IS COMPLETE,THE
C     ..............SOLUTION CAN NOW BE OBTAINED FOR ANY RHS
C     ..............OF AA*XX=RHS.
      ENTRY CS009B(RHS,XX)
C     ..............THIS ENTRY POINT IS USED WHEN THERE IS
C     ..............MORE THAN ONE RHS AND FOR OBTAINING THE
C     ..............INVERSE.
   66 FIN=0
C     ..............K IS AN ITERATION COUNTER
C     ..............XX  ITERATED SOLUTIONS
```

```
C       ..............RR  ITERATED RESIDUALS
C       ..............BEGIN WITH STARTING VALUES FOR XX & RR
C       ..............OF THE NULL VECTOR AND B RESPECTIVELY
        DO 14 I=1,N
        XX(I)=0.0
   14   RR(I)=RHS(I)
        K=0
C       ..............EXECUTE FORWARD SUBSTITUTION AND OBTAIN
C       ..............SOLUTION TO LL*Y=P*RR
  141   IF(K.GT.1) GO TO 3000
        IF(K.EQ.0) GO TO 146
        SCALE=ABS(RR(1))
        DO 142 I=2,N
  142   SCALE=AMIN1(SCALE,ABS(RR(I)))
        FACTOR=1.0
        IF(SCALE.LE.1.0E-10) SCALE=1.0
  143   IF(SCALE.GE.1.0) GO TO 144
        FACTOR=FACTOR*10.0
        SCALE=SCALE*10.0
        GO TO 143
  144   DO 145 I=1,N
  145   RR(I)=RR(I)*FACTOR
  146   DO 18 I=1,N
        J=1
        SS=DBLE(RR(PIVCT(I)))
        RR(PIVCT(I))=RR(I)
   16   CONTINUE
        IF(J.EQ.I) GO TO 17
        SS=SS-(DBLE(BB(I,J))*DBLE(RR(J)))
        J=J+1
        GO TO 16
   17   RR(I)=SNGL(ROUND+(SS/DBLE(BB(I,I))))
   18   CONTINUE
C       ..............BACK SUBSTITUTION FOR UU*RR=Y
C       ..............K-TH CORRECTION TERMS ARE OVERWRITTEN ON RR
        I=N
   19   J=I
        SS=DBLE(RR(I))
   20   IF(J.EQ.N) GO TO 21
        J=J+1
        SS=SS-(DBLE(BB(I,J))*DBLE(RR(J)))
        GO TO 20
   21   RR(I)=SNGL(SS+ROUND)
        IF(I.EQ.1) GO TO 22
        I=I-1
        GO TO 19
C       ..............CHEBYSHEV NORM FOR XX AND RR
   22   IF(K.EQ.0) GO TO 222
        DO 221 I=1,N
  221   RR(I)=RR(I)/FACTOR
  222   NORMXX=0.0
        NORMEE=0.0
        DO 23 I=1,N
```

```
      NORMXX=AMAX1(NORMXX,ABS(XX(I)))
   23 NORMEE=AMAX1(NORMEE,ABS(RR(I)))
C     ..............BEGIN ITERATIVE CORRECTION PROCEEDURE
C     ..............ONLY ONE CORRECTION STEP IS PERFORMED.
C     ..............IF WARRENTED STATEMENT 141 CAN BE ALTERED
C     ..............TO PERMIT ADDITIONAL CORRECTIONS.
      IF(K.EQ.0) GO TO 29
      IF(K.EQ.1) GO TO 25
   24 IF((NORMEE/NORMXX).LT.2.0**(-23)) GO TO 28
      GO TO 29
   25 IF((NORMEE/NORMXX).LE.0.5) GO TO 24
      GO TO 2000
C     ..............FIN=1 INDICATES ACCEPTANCE OF SOLUTION
   28 FIN=1
C     ..............ADDITION OF THE CORRECTION TERMS
C     ..............K-TH ITERATED SOLUTION IS XX
   29 DO 30 II=1,N
   30 XX(II)=XX(II)+RR(II)
C     ..............ACCUMULATION OF RESIDUALS
      DO 32 I=1,N
      SS=DBLE(RHS(I))
      DO 31 J=1,N
   31 SS=SS-(DBLE(AA(I,J))*DBLE(XX(J)))
   32 RR(I)=SNGL(SS+ROUND)
      K=K+1
      IF(FIN.NE.1) GO TO 141
      RETURN
C                    ERROR CONDITIONS AND RETURNS
C
 1000 WRITE(6,1001)
 1001 FORMAT('****** ERROR CONDITION- SINGULAR OR NEAR SINGULAR MATRIX *
     1*****')
      BABY=BABY+1.0
      RETURN
 2000 WRITE(6,2001)
 2001 FORMAT('****** ERROR CONDITION-MATRIX IS SINGULAR TO WORKING ACCUR
     1ACY ******')
      BABY=BABY+1.0
 3000 RETURN
      END
```

APPENDIX G

Formula to obtain the 'first horizontal derivative' with respect to x.

$$g'(x) = \frac{2\pi i}{N} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} m \; A_{m,n} \; e^{\frac{2\pi i(mx+ny)}{N}}$$

where

$A_{m,n}$ is complex and represents the FOURIER coefficients of the observed field.

FORTRAN IV

PROGRAM TO OBTAIN 'FIRST HORIZONTAL DERIVATIVE' USING FAST FOURIER TRANSFORM SUBROUTINES.

DELT	.....	Digital interval of the data set
N(1)	.....	Size of the data array along x-axis
N(2)	.....	Size of the data array along y-axis (Normally considered a square matrix)
N(3)	.....	is set to zero
X1(N(1),N(2))	.....	The digitized data as reduced after filtering and decimation is stored in this vector.

READ IN THE FOLLOWING PARAMETERS

1.	DELT	according to FORMAT(5X,F10.5)
2.	N(1),N(2),N(3)	according to FORMAT(5X,3I5)
3.	X1(N(1),N(2))	according to FORMAT(1X,6E11.4)

```
C       CALCULATIONS FOR  'FIRST HORIZONTAL DERIVATIVE'
C
C       THIS PROGRAM CALCULATES THE FIRST HORIZONTAL DERIVATIVE, USING
C       ANY POTENTIAL FIELD DATA.
C
C       THE PROGRAM USES TWO DIMENSIONAL FOURIER TRANSFORM WHICH IS BASED
C       ON THE ALGORITHM DEVELOPED BY GOOD (1958) AND MODIFIED BY COOLEY-
C       TUKEY (1966) AND GENTLEMAN-SANDE (1966).
C
C       X1(I,J) IS THE REAL INPUT MATRIX IN SPACE DOMAIN.
C
C       X(J) IS THE FINAL SYMMETERIZED DATA, USING CYCLIC PROPORTIES OF
C       FOURIER TRANSFORMS.
C
C       FOR COMPUTATION PROGRAM USES X(J)+IY(J) WHERE Y(J)=0.0
C       OUTPUT RESULTS ARE GIVEN AS X(J)+IY(J) IN FREQUENCY DOMAIN
C       N(J,K,0)  ***  ARRAY OF DATA IS TWO DIMENSIONAL WITH SIDE J BY K.
C       NOPTS   ***   NUMBER OF DATA POINTS  (J*K).
C       DELT   ***   THE DIGITIZING INTERVAL IN MILES.
C
        DIMENSION  X(15000),Y(15000),S(15000),N(3),X1(60,60)
     20 FORMAT (5X,3I5)
    22  FORMAT(1H0,31X,14HDECLINATION = ,F10.5,23X,24HSIZE OF OUTPUT MATRI
       1X = ,I5)
     30 FORMAT (5X,F10.5)
    34  FORMAT(5X,8F5.0)
     35 FORMAT (1X,6E11.4)
     36 FORMAT(1X,10E13.4)
     40 FORMAT(1H1,60X,10HINPUT DATA)
     42 FORMAT(1H ,60X,10H**********)
     45 FORMAT (1H0)
    46  FORMAT(1H1)
     50 FORMAT(1H ,25X,80(1H*))
     60 FORMAT(1X,6HDELT =,F8.3,8X,18HNYQUIST FREQUENCY=,F8.3)
     70 FORMAT(1H0,22X,15HFREQUENCY =2*J*,F10.6,1X,15HCYCLES PER MILE)
   316  FORMAT(1H1,40X,53HFIRST HORIZONTAL DERIVATIVE ALONG THE DECLINATIO
       1N '0')
   317  FORMAT(41X,53(1H*))
    320 FORMAT(1HL,56X,17HREAL (SPACE) PART)
    325 FORMAT(1H ,56X,17H*****************)
    261 FORMAT(1H1,55X,21HIMAGINARY(SPACE) PART)
        READ (5,30) DELT
        READ (5,20) (N(J),J=1,3)
        NOPTS = N(1)*N(2)
        NN=N(1)
        NNN=N(2)
        DO 9 I=1,NN
        READ(5,35)  (X1(I,J),J=1,NNN)
      9 CONTINUE
        WRITE (6,40)
        WRITE(6,42)
        DO 10 I=1,NN
        WRITE(6,36) (X1(I,J),J=1,NNN)
```

```
      WRITE (6,45)
   10 CONTINUE
      WRITE (6,50)
C
C     SETTING X AS THE FINAL SYMMETERIZED MATRIX OF SIZE (2N-2)*(2N-2)
C     REMEMBER HERE X IS STORED IN A VECTOR FORM.
      DO 1 I=1,NN
      DO 2 J=1,NNN
      J1=(2*NN-2)*(I-1)+J
      X(J1)=X1(I,J)
    2 CONTINUE
      N1=NN-2
      DO 3 J=1,N1
      J2=J1+J
      X(J2)=X1(I,NN-J)
    3 CONTINUE
   1  CONTINUE
      N2=2*NN-2
      DO 5 K=1,N1
      DO 4 J=1,N2
      J3=J2+J
      X(J3)=X(J3-(4*NN-4)*K)
    4 CONTINUE
      J2=J3
    5 CONTINUE
C     END OF SYMMETERIZED VECTOR.
C     NOW WE RESET N(1),N(2),NOPTS,ETC.
      N(1)=2*NN-2
      N(2)=2*NNN-2
      NN=N(1)
      NNN=N(2)
      NOPTS=N(1)*N(2)
      FN = 1./(2.*DELT)
      WRITE (6,60) DELT,FN
      FP = FN/(FLOAT(NOPTS))
      WRITE (6,70) FP
      WRITE(6,46)
      CALL  WRIT(X,NN,NNN)
      CALL  ZERO(Y,NOPTS)
      CALL  ZERO(S,NOPTS)
      CALL  ARMDFT(N,X,Y,S)
C
      WRITE(6,46)
      WRITE(6,320)
      CALL  WRIT(X,NN,NNN)
      WRITE(6,261)
      CALL  WRIT(Y,NN,NNN)
C     TO MULTIPLY FREQUENCIES BY A FACTOR
C     2.0*3.14159*J/NN   , FOR I=0,1,2,....NN  AND J=0,1,2,...NNN
C
      DO 121 I=1,NN
      IF(I.GT.(NN/2+1)) GO TO 11
      A=I-1
```

```
      GC TC 12
   11 A=-(NN+1-I)
   12 CONTINUE
      K=(I-1)*NNN+1
      L=K+NNN-1
      DO 121 J=K,L
      IF(J.GT.((L+K)/2+1)) GC TO 13
      B=J-1-(NN*(I-1))
      GC TC 14
   13 B=-(NN+1-J+(NN*(I-1)))
   14 CONTINUE
      CCNT=(2.0*3.14159*B)/NN
      XTEMP=X(J)
      YTEMP=Y(J)
      X(J)= -1.0*YTEMP*CCNT
      Y(J)=XTEMP*CCNT
  121 CONTINUE
C     SET THE LAST POINT (I.E. NN/+1) OF THE
C     IMIGINARY VECTOR TO -ZERO.
      K=1
      L=0
      DO 122 M=1,NN
      L=L+NNN
      NSET=(K+L)/2+1
      Y(NSET)=-0.0
      K=L+1
 122  CONTINUE
C
      WRITE(6,46)
      WRITE(6,320)
      CALL  WRIT(X,NN,NNN)
      WRITE(6,261)
      CALL  WRIT(Y,NN,NNN)
      J=0
      DO 250 J3=1,NOPTS
      J=J+1
C     REPLACE FOURIER COEFFICIENTS BY COMPLEX  CONJUGATE.
      F=Y(J)
      Y(J)=-F
  250 CONTINUE
      CALL  ZERO(S,NOPTS)
      CALL  ARMDFT(N,X,Y,S)
      J=0
      DO 350 J4=1,NOPTS
      J=J+1
C     TAKE COMPLEX CONJUGATE.
      GG=Y(J)
      Y(J)=-GG
  350 CONTINUE
      J=0
      DO 400 J5=1,NOPTS
      J=J+1
      X(J)=X(J)/FLOAT(NOPTS)
```

```
      Y(J)=Y(J)/FLOAT(NCPTS)
  400 CONTINUE
C
      WRITE(6,46)
      WRITE(6,320)
      CALL  WRIT(X,NN,NNN)
      WRITE(6,261)
      CALL  WRIT(Y,NN,NNN)
C     RESET NN AND NNN . AND PICK UP USEFUL SIZE OF THE FINAL RESULTS.
C     WE ALSO OBTAIN PUNCHED DECK OF THE FINAL RESULTS.
C
      WRITE(6,316)
      WRITE(6,317)
      WRITE (6,320)
      WRITE (6,325)
      NN=(NN+2)/2
      NNN=(NNN+2)/2
      DO 7 I=1,NN
      DO 8 J=1,NNN
      X1(I,J)=X((2*NN-2)*(I-1)+J)
C
    8 CONTINUE
    7 CONTINUE
      DO 21 I=1,NN
      WRITE(6,36) (X1(I,J),J=1,NNN)
      WRITE(6,45)
   21 CONTINUE
      WRITE (6,50)
      STOP
      END
```

APPENDIX H

Modified formula for the reduction of total magnetic field to the pole:

$$T_R = \frac{1}{\sin I_{0\nu} \sin I_\nu} \sum_m \sum_n A_{mn} \frac{p^4_{mn} - \psi_1\psi_2 p^2_{mn} - i(\psi_1+\psi_2)p^3_{mn}}{(p^2_{mn}-\psi_1\psi_2)^2+(\psi_1+\psi_2)^2 p^2_{mn}} \times$$

$$e^{i(k_m x+k_n y)}\, e^{p_{mn} z}$$

Where

A_{mn} is complex and represents the FOURIER coefficients of the observed fields

$k_m = \frac{2\pi m}{L}$

$k_n = \frac{2\pi n}{L}$

$p_{mn} = \frac{2\pi}{L}(m^2+n^2)^{\frac{1}{2}}$

L is the wave length in the x and y direction

$I_{0\nu} = 180 - I_0$

$I_\nu = 180 - I$

$\psi_1 = (k_m \cos D_0 + k_n \sin D_0) \cot I_{0\nu}$

$\psi_2 = (k_m \cos D + k_n \sin D) \cot I_\nu$

I_0, D_0 are the inclination and declination of the polarizing vector of the rocks

I,D are the inclination and declination of the earth's normal field.

FORTRAN IV

PROGRAM TO REDUCE THE TOTAL MAGNETIC FIELD TO THE 'POLE', USING FAST FOURIER TRANSFORM SUBROUTINES.

DELT	....	Digital interval of the data set
AI0	....	Inclination of the polarizing vector of the rocks, in degrees
AI	....	Inclination of the earth's normal field, in degrees
D0	....	Declination of the polarizing vector of rocks, in degrees
D	....	Declination of the earth's normal field in degrees
N(1)	....	Size of the data array along x-axis
N(2)	....	Size of the data array along y-axis (normally considered a square matrix)
N(3)	....	is set to zero
X1(N(1),N(2))	...	The digitized data as reduced after filtering and decimation is stored in this vector

READ IN THE FOLLOWING PARAMETERS

1. DELT according to FORMAT(5X,5F10.5)
2. AI0,AI,D0,D according to FORMAT(5X,5F10.5)
3. N(1),N(2),N(3) according to FORMAT(5X,3I5)
4. X1(N(1),N(2)) according to FORMAT(1X,6E11.4)

```
C     THIS PROGRAM PROVIDES THE 'MAGNETIC ANOMALIES REDUCED TO THE POLE'
C
C     THE METHOD WAS ORIGINALLY PROPOSED BY BARANOV (1957), AND
C     MODIFIED BY BHATTACHARYA (1965). IN THE PRESENT PROGRAM,
C     USING BHATTACHARYA'S APPROACH ,THE FORMULA'S HAVE BEEN DERIVED
C     TO MAKE USE OF THE FAST FOURIER TRANSFORM ALGORITHM.
C
C     THE INPUT DATA CONSISTS OF THE 'TOTAL MAGNETIC FIELD VALUES'.
C
C
C     AIO= INCLINATION OF THE POLARIZING VECTOR OF ROCKS, IN DEGREES.
C     DO = DECLINATION OF THE POLARIZATION VECTOR OF ROCKS, IN DEGREES.
C     AI = INCLINATION OF THE EARTH'S NORMAL FIELD, IN DEGREES.
C     D  = DECLINATION OF THE EARTH'S NORMAL FIELD, IN DEGREES.
C
C
C        THE PROGRAM USES TWO DIMENSIONAL FOURIER TRANSFORM WHICH IS BASED
C        ON THE ALGORITHM DEVELOPED BY GOOD (1958) AND MODIFIED BY COOLEY-
C        TUKEY (1965) AND GENTLEMAN-SANDE (1966).
C
C        X1(I,J) IS THE REAL INPUT MATRIX IN SPACE DOMAIN.
C
C        X(J) IS THE FINAL SYMMETERIZED DATA, USING CYCLIC PROPORTIES OF
C        FOURIER TRANSFORMS.
C
C        FOR COMPUTATION PROGRAM USES X(J)+IY(J) WHERE Y(J)=0.0
C        OUTPUT RESULTS ARE GIVEN AS X(J)+IY(J) IN FREQUENCY DOMAIN
C        N(J,K,0)  ***  ARRAY OF DATA IS TWO DIMENSIONAL WITH SIDE J BY K.
C        NOPTS    ***    NUMBER OF DATA POINTS  (J*K).
C        DELT    ***   THE DIGITIZING INTERVAL IN MILES.
C
      DIMENSION  X(10000),Y(10000),S(10000),N(3),X1(60,60)
   20 FORMAT (5X,3I5)
   30 FORMAT (5X,5F10.5)
   99 FORMAT(10X,5F10.0)
   34 FORMAT(1X,13F5.2)
   35 FORMAT (1X,6E11.4)
   36 FORMAT(1X,10E13.4)
   40 FORMAT(1H1,60X,10HINPUT DATA)
   42 FORMAT(1HT,60X,10H**********)
   45 FORMAT (1H0)
   46 FORMAT(1H1)
   50 FORMAT(1HT,25X,80(1H*))
   60 FORMAT(1X,6HDELT =,F8.3,6X,18HNYQUIST FREQUENCY=,F8.3)
   70 FORMAT(1H0,22X,15HFREQUENCY =2*J*,F10.6,1X,15HCYCLES PER MILE)
  316 FORMAT(1H1,40X,'MAGNETIC ANOMALIES ''REDUCED TO THE POLE''')
  317 FORMAT(41X,47(1H*))
  320 FORMAT(1HL,56X,17HREAL (SPACE) PART)
  325 FORMAT(1HT,56X,17H*****************)
      READ (5,30) DELT
      READ(5,30)  AIO,AI,DO,D
      READ (5,20) (N(J),J=1,3)
      NOPTS = N(1)*N(2)
```

```
      NN=N(1)
      NNN=N(2)
      DO 9 I=1,NN
      READ(5,35)  (X1(I,J),J=1,NNN)
    9 CONTINUE
      WRITE (6,40)
      WRITE(6,42)
      DO 10 I=1,NN
      WRITE(6,36) (X1(I,J),J=1,NNN)
      WRITE (6,45)
   10 CONTINUE
      WRITE (6,50)
C     SETTING X AS THE FINAL SYMMETERIZED MATRIX OF SIZE (2N-2)*(2N-2)
C     REMEMBER HERE X IS STORED IN A VECTOR FORM.
      DO 1 I=1,NN
      DO 2 J=1,NNN
      J1=(2*NN-2)*(I-1)+J
      X(J1)=X1(I,J)
    2 CONTINUE
      N1=NN-2
      DO 3 J=1,N1
      J2=J1+J
      X(J2)=X1(I,NN-J)
    3 CONTINUE
    1 CONTINUE
      N2=2*NN-2
      DO 5 K=1,N1
      DO 4 J=1,N2
      J3=J2+J
      X(J3)=X(J3-(4*NN-4)*K)
    4 CONTINUE
      J2=J3
    5 CONTINUE
C     END OF SYMMETERIZED VECTOR.
C     NOW WE RESET N(1),N(2),NOPTS,ETC.
      N(1)=2*NN-2
      N(2)=2*NNN-2
      NN=N(1)
      NNN=N(2)
      NOPTS=N(1)*N(2)
      FN = 1./(2.*DELT)
      WRITE (6,60) DELT,FN
      FP = FN/(FLOAT(NOPTS))
      WRITE (6,70) FP
C     REMOVE D.C.
C
C
      CALL  ZERO(Y,NOPTS)
      CALL  ZERO(S,NOPTS)
      CALL  ARMDFT(N,X,Y,S)
C
      WRITE(6,46)
      CALL  WRIT(X,NN,NNN)
```

```
      WRITE(6,46)
      CALL  WRIT(Y,NN,NNN)
      CALL  ZERO(Y,NOPTS)
      TWOPI=6.283185307
      CONST=TWOPI/360.0
C     CONVERSION FROM DEGREES TO RADIANS.
      DO=DO*CONST
      D=D*CONST
      AIO=AIO*CONST
      AI=AI*CONST
      COTR=COTAN(AIO)
      COTE=COTAN(AI)
      COSDR=COS(DO)
      COSDE=COS(D)
      SINDR=SIN(DO)
      SINDE=SIN(D)
      SINIR=SIN(AIO)
      SINIE=SIN(AI)
      SIOSI=1./(SINIR*SINIE)
C     TO MULTIPLY FREQUENCIES BY A FACTOR
C
      DO 121 I=1,NN
      IF(I.GT.(NN/2+1)) GO TO 11
      A=I-1
      GO TO 12
   11 A=-(NN+1-I)
   12 CONTINUE
      K=(I-1)*NNN+1
      L=K+NNN-1
      DO 121 J=K,L
      IF(J.GT.((L+K)/2+1)) GO TO 13
      B=J-1-(NN*(I-1))
      GO TO 14
   13 B=-(NN+1-J+(NN*(I-1)))
   14 CONTINUE
      IF(A.EQ.0.0.AND.B.EQ.0.0)  GO TO 121
      PMN=SQRT(A**2+B**2)
      AKM=A/PMN
      BKN=B/PMN
      SAI1=COTR*(AKM*COSDR+BKN*SINDR)
      SAI2=COTE*(AKM*COSDE+BKN*SINDE)
      XTEMP=X(J)
      YTEMP=Y(J)
      DEN=1./((1+SAI1**2)*(1+SAI2**2))
      FPART=DEN*(1-SAI1*SAI2)
      SPART=DEN*(SAI1+SAI2)
      X(J)=FPART*XTEMP-SPART*YTEMP
      Y(J)=FPART*YTEMP+SPART*XTEMP
C
C     MULTIPLY WITH THE 'INCLINATION FACTOR' DUE TO THE ROCKS AND THE
C     EARTH'S MAGNETIC FIELD.
      X(J)=X(J)*SIOSI
      Y(J)=Y(J)*SIOSI
```

```
  121 CONTINUE
C
C      SET D.C. OF REAL PART TO ZERO
       X(1)=0.0
C
C
       WRITE(6,46)
       CALL  WRIT(X,NN,NNN)
       WRITE(6,46)
       CALL  WRIT(Y,NN,NNN)
       J=0
       DO 250 J3=1,NCPTS
       J=J+1
C      REPLACE FOURIER COEFFICIENTS BY COMPLEX  CONJUGATE.
       F=Y(J)
       Y(J)=-F
  250 CONTINUE
       CALL  ZERO(S,NCPTS)
       CALL  ARMDFT(N,X,Y,S)
       J=0
       DO 350 J4=1,NCPTS
       J=J+1
C      TAKE COMPLEX CONJUGATE.
       GG=Y(J)
       Y(J)=-GG
  350 CONTINUE
       J=0
       DO 400 J5=1,NCPTS
       J=J+1
       X(J)=X(J)/FLOAT(NCPTS)
       Y(J)=Y(J)/FLOAT(NCPTS)
  400 CONTINUE
       WRITE(6,46)
       CALL  WRIT(X,NN,NNN)
       WRITE(6,46)
       CALL  WRIT(Y,NN,NNN)
C      RESET NN AND NNN . AND PRINT THE SIZE OF THE FINAL RESULTS.
C
C
       WRITE(6,316)
       WRITE(6,317)
       WRITE (6,320)
       WRITE (6,325)
       NN=(NN+2)/2
       NNN=(NNN+2)/2
       DO 7 I=1,NN
       DO 8 J=1,NNN
       X1(I,J)=X((2*NN-2)*(I-1)+J)
C
    8 CONTINUE
    7 CONTINUE
       CALL  PLOTER(X1,NN)
       DO 21 I=1,NN
```

```
      WRITE(6,36) (X1(I,J),J=1,NNN)
      WRITE(7,35)  (X1(I,J),J=1,NNN)
      WRITE(6,45)
   21 CONTINUE
      WRITE (6,50)
C
      STOP
      END
```

```
      SUBROUTINE  PLOTER(Z,NN)
      DIMENSION  X(60),Y(60),Z(60,60),WORK(1000)
      CALL  PLOTS(WORK(1),4000)
      DO 1 I=1,NN
      DO 2 J=1,NN
      X(J)=4.0+J
      Y(J)=26.0-I
      IF(Y(J).LT.1.0)  GO TO 11
      X(J)=44.0+J
      Y(J)=54.0-I
11    CONTINUE
      X1=X(J)-0.15
      Y1=Y(J)+0.1
      CALL  SYMBOL(X(J),Y(J),0.05,4,0.0,-1)
      CALL  NUMBER(X1,Y1,0.1,Z(I,J),0.0,-1)
 2    CONTINUE
 1    CONTINUE
      CALL  PLOT(40.0,0.0,-3)
      CALL  PLOT(0.0,0.0,999)
      RETURN
      END
```

```
      SUBROUTINE ZERO(X,NOPTS)
C     SETS THE VECTOR TO ZERO
      REAL X(10)
      DO 1 J=1,NOPTS
      X(J)=0.0
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE WRIT(AX,NN,NNN)
C     TO PRINT OUT THE RESULTS
      REAL AX(10)
   36 FORMAT(1X,10E15.4)
   45 FORMAT(1HJ)
      K=1
      L=0
      DO 1 M=1,NN
      L=L+NNN
      WRITE(6,36) (AX(J),J=K,L)
      WRITE(6,45)
      K=L+1
    1 CONTINUE
      RETURN
      END
```

APPENDIX I

Computation of cross-power and coherence function

Coherence function is obtained by using equation (4.3) described in Chapter IV, i.e.

$$\mathrm{Coh}^2_{Z_O,Z_T}(k_x,k_y) = \frac{\left| P_{Z_O,Z_T}(k_x,k_y) \right|^2}{P_{Z_O}(k_x,k_y)\; P_{Z_T}(k_x,k_y)} \qquad \text{.....(I-1)}$$

The cross-correlation between the two data sets Z_O and Z_T provides the raw cross-power spectra. The cross-correlation is given by

$$C_{Z_O,Z_T}(k_x,k_y) = c(k_x,k_y) + i\, q(k_x,k_y) \qquad \text{.....(I-2)}$$

where

$c(k_x,k_y)$ is known as co-spectrum and

$q(k_x,k_y)$ is the quadrature spectrum

The raw cross-power spectra $P_r(k_x,k_y)$ is given by

$$P_r(k_x,k_y) = \sqrt{\{c^2(k_x,k_y) + q^2(k_x,k_y)\}} \qquad \text{.....(I-3)}$$

The convolution of equation (I-3) with the two-dimensional Hamming window provides the smooth cross-power spectra $P_{Z_O,Z_T}(k_x,k_y)$.

FORTRAN IV

PROGRAM TO CALCULATE CO-SPECTRUM, QUADRATURE SPECTRUM AND COHERENCE FUNCTION COEFFICIENTS.

N4	...	Size of the square data matrix
LF	...	Size of the two-dimensional window
N(1) = N4		
N(2) = N4		For a square matrix
N(3) = 0		
NTAO	...	Number of lags considered. Normally taken max. lags which is equal to N(1) or N(2)
DELT	...	The digital spacing of the data
A(N4,N4)	...	To store one set of digitized field data
B(N4,N4)	...	To store second set of digitized field data
F(LF,LF)	...	To store filter coefficients of two-dimensional window

READ IN THE FOLLOWING PARAMETERS

1. N4,LF according to FORMAT(5X,4I5)
2. N(1),N(2),N(3) according to FORMAT(5X,4I5)
3. NTAO according to FORMAT(5X,4I5)
4. DELT according to FORMAT(5X,F10.6)
5. A(N4,N4) according to FORMAT(1X,6E11.4)
6. B(N4,N4) according to FORMAT(1X,6E11.4)
7. F(LF,LF) according to FORMAT(1X,6E11.4)

```
C     TWO DIMENSIONAL POWER SPECTRA AND COHERENCY TEST.
C
C     THIS PROGRAM OBTAINS THE TWO-DIMENSIONAL AUTO- AND CROSS-POWER
C     SPECTRA OF THE TWO GIVEN DATA SETS.
C
C     RAW SPECTRAL ESTIMATES ARE SMOOTHED BY USING A TWO DIMENSIONAL
C     HAMMING WINDOW IN THE FREQUENCY DOMAIN .
C
C
C     THE PROGRAM MAKES USE OF THE TWO-DIMENSIONAL FOURIER TRANSFORMS,
C     BASED ON THE ALGORITHM OF GOOD(1958) AS MODIFIED BY COOLEY-TUKEY
C     (1966) AND GENTLEMAN-SANDE(1966)
C
C     FOR CROSS-POWER, FIRST SET OF DATA IS READ IN 'BY' VECTOR
C     WHICH CONSISTS OF SIZE N(1)*N(2), AND THE DATA WITH WHICH IT IS
C     TO BE CORRELATED IS ENTERED IN VECTOR 'DY' WHICH IS OF THE SAME
C     SIZE AS 'BY'
C
C     N(J,K,L)    ***   ARRAY OF DATA IS TWO-DIMENSIONAL WITH SIZE J BY
C
C     N4    ***   SIZE OF THE DATA MATRIX
C
C     DELT     ***   THE DIGITIZING INTERVAL IN MILES
C
C     NOPTS        ***   NUMBER OF DATA POINTS (J*K)=N(1)*N(2)
C
C     N TAU        ***   NUMBER OF LAGS THAT ALSO MEANS THAT WE ADD
C                        SO MANY ZEROS. MAXIMUM LAGS CAN BE EQUAL TO
C                        THE DIMENSION N(1) OR N(2) OF THE DATA POINTS
C                        READ IN.
C     LF  ..  SIZE OF THE TWO DIMENSIONAL FILTER.
C     ISS  ..  WRITES REAL AND IMAGINARY COEFFICIENTS OF THE
C              TWO SETS OF DATA.
C
      DIMENSION  A(26,26),B(26,26),C(26,26),D(26,26),COH(26,26),F(5,5),
     1BY(3000),DY(3000)
C
C     ANY CHANGE OF DIMENSIONS IN THE COMMON CARDS ALSO GOES IN THE
C     SUBROUTINE 'CROSS'. IN SUBROUTINE 'CROSS' CARD IDIME=3000 WOULD
C     ALSO NEED TO BE CHANGED.
C
      COMMON  AX(3000),CX(3000),S(3000),X(3000),Y(3000)
      COMMON  DELT
      COMMON  N(3),ISS
      COMMON  NN,NNN,NOPTS,NTAU
      REAL  AX,BY,CX,DY,S,X,Y,DELT
      INTEGER  N,NN,NNN,NOPTS,NTAU
    7 FORMAT( 46X,'CROSS_CORRELATION OF THE INPUT DATA'  )
    8 FORMAT(1H1,45X,35(1H*))
   10 FORMAT(5X,4I5)
   20 FORMAT(5X,F10.6)
   29 FORMAT(5X,I5,2F10.5)
   30 FORMAT(1X,4E16.8)
```

```
   35   FORMAT(1X,6E11.4)
   36   FORMAT(1X,10E13.4)
   45   FORMAT(1H0)
   46   FORMAT(1H1)
   50   FORMAT(1HT,25X,80(1H*))
  601   FORMAT(1H1,60X,'INPUT DATA'    )
  602   FORMAT(1HT,60X,10(1H*))
  603   FORMAT(60X,'O/RHO RATIO')
  605   FORMAT(50X,'COHERENCE FUNCTION COEFFICIENTS')
        READ(5,10) N4,LF,ISS
        READ(5,10) (N(J),J=1,3)
        READ(5,10) NTAC
        READ(5,20) DELT
        NOPTS=N(1)*N(2)
        NN=N(1)
        NNN=N(2)
C
C       READ IN THEORETICALLY CALCULATED DATA.
        DO 1 I=1,N4
        READ(5,35)  (A(I,J),J=1,N4)
    1   CONTINUE
        DO 13 I=1,N4
        DO 13 J=1,N4
        A(I,J)=A(I,J)/(0.667E-07)
   13   CONTINUE
C       READ IN OBSERVED MAGNETIC FIELD 'REDUCED TO THE POLE'.
        DO 2 I=1,N4
        READ(5,35)  (B(I,J),J=1,N4)
    2   CONTINUE
C       READ IN TWO-DIMENSIONAL FILTER COEFFICIENTS.
        DO 3 I=1,LF
        READ(5,35)  (F(I,J),J=1,LF)
    3   CONTINUE
C
C
C       INITIALLY WE TRANSFER THE DATA IN THE IMAG. VECTORS 'BY' AND 'DY'
C       FROM WHICH WE WILL TRANSFER TO 'AX' AND 'CX' AFTER ADDING THE
C       NECESSARY ZEROS FOR CROSS-CORRELATION.
C
        CALL  ZERO(DY,NOPTS)
        CALL  TRANSF(DY,A,NN)
C
        CALL  ZERO(BY,NOPTS)
        CALL  TRANSF(BY,B,NN)
C
C       NOW WE CALCULATE CROSS-POWER SPECTRA
C
        CALL  CROSS(BY,DY)
C
        DO 14 I=1,N4
        DO 14 J=1,N4
        C(I,J)=AX((2*N4)*(I-1)+J)
   14   CONTINUE
```

```
      WRITE(6,46)
      WRITE(6,603)
      WRITE(6,45)
      WRITE(6,45)
      DO 16 I=1,N4
      WRITE(6,36)  (C(I,J),J=1,N4)
      WRITE(7,35)  (C(I,J),J=1,N4)
      WRITE(6,45)
   16 CONTINUE
      CALL  PLOTER(C,N4)
C
      DO 4 I=1,N4
      DO 4 J=1,N4
      C(I,J)=SQRT(X((2*N4)*(I-1)+J)**2+Y((2*N4)*(I-1)+J)**2)
    4 CONTINUE
C
C     CONVOLVE 'RAW' CROSS-POWER WITH TWO-DIMENSIONAL HAMMING WINDOW
C     IN FREQUENCY DOMAIN TO OBTAIN SMOOTH CROSS-POWER SPECTRA.
C
      CALL  CONV 2D(C,F,N4,LF)
C
C     OBTAIN SQUARE OF CROSS-POWER SPECTRA FOR COMPUTING THE COHERENCE
C     FUNCTION.
C
      NT=N4-LF+1
C     C(I,J) IS REDUCED TO NT BY NT MATRIX SIZE.
      DO 6 I=1,NT
      DO 6 J=1,NT
      C(I,J)=C(I,J)*C(I,J)
    6 CONTINUE
      NN=NN-NTAO
      NNN=NNN-NTAO
      N(1)=NN
      N(2)=NNN
      NOPTS=N(1)*N(2)
      ISS=ISS+1
C     TO CALCULATE AUTO-SPECTRA OF MAGNETIC DATA.
      CALL  ZERO(DY,NOPTS)
      CALL  TRANSF(DY,A,NN)
      CALL  ZERO(BY,NOPTS)
      CALL  TRANSF(BY,A,NN)
      CALL  CROSS(BY,DY)
      DO 11 I=1,N4
      DO 11 J=1,N4
      D(I,J)=X((2*N4)*(I-1)+J)
   11 CONTINUE
C
C     CONVOLVE 'RAW' AUTO-POWER WITH TWO-DIMENSIONAL HAMMING WINDOW
C     IN FREQUENCY DOMAIN TO OBTAIN SMOOTH AUTO-POWER SPECTRA.
C
      CALL  CONV 2D(D,F,N4,LF)
      NN=NN-NTAO
      NNN=NNN-NTAO
```

```
      N(1)=NN
      N(2)=NNN
      NCPTS=N(1)*N(2)
C     TO CALCULATE AUTO-SPECTRA OF GRAVITY DATA.
      CALL  ZERO(DY,NCPTS)
      CALL  TRANSF(DY,E,NN)
      CALL  ZERO(BY,NCPTS)
      CALL  TRANSF(BY,B,NN)
      CALL  CROSS(BY,DY)
      DO 12 I=1,N4
      DO 12 J=1,N4
      A(I,J)=X((2*N4)*(I-1)+J)
   12 CONTINUE
C
C     CONVOLVE 'RAW' AUTO-POWER WITH TWO-DIMENSIONAL HAMMING WINDOW
C     IN FREQUENCY DOMAIN TO OBTAIN SMOOTH AUTO-POWER SPECTRA.
C
      CALL  CONV 2D(A,F,N4,LF)
      NN=NN-NTAU
      NNN=NNN-NTAU
      N(1)=NN
      N(2)=NNN
      NCPTS=N(1)*N(2)
C     CALCULATE COHERENCE FUNCTION COEFFICIENTS.
      DO 5 I=1,NT
      DO 5 J=1,NT
      COH(I,J)=C(I,J)/(D(I,J)*A(I,J))
    5 CONTINUE
      WRITE(6,46)
      WRITE(6,605)
      WRITE(6,45)
      WRITE(6,45)
      DO 15 I=1,NT
      WRITE(6,36)  (COH(I,J),J=1,NT)
      WRITE(6,45)
   15 CONTINUE
      STOP
      END
```

```
      SUBROUTINE  CROSS(BY,DY)
C
C     THIS SUBROUTINE CALCULATES POWER SPECTRA
C
C     ANY CHANGE OF DIMENSION CARD NEED ALSO A CHANGE IN THE
C     IDIME=3000 CARD.
C
      DIMENSION  BY(3000),DY(3000)
      COMMON  AX(3000),CX(3000),S(3000),X(3000),Y(3000)
      COMMON  DELT
      COMMON  N(3),ISS
      COMMON  NN,NNN,NCPTS,NTAC
      REAL  AX,BY,CX,DY,S,X,Y,DELT
      INTEGER  N,NN,NNN,NCPTS,NTAC
C
C     FIRST WE SET ALL VALUES OF THE 'AX' AND 'CX'- ARRAY EQUAL TO ZERO
C
      IDIME=3000
      CALL ZERO(AX,IDIME)
      CALL ZERO(CX,IDIME)
C
C     THE MATICES 'BY' AND 'DY' ARE IN THE NORTHWEST QUADRANTS,
C     FROM WHICH WE WISH TO FORM LARGE MATRIX 'AX' AND 'CX' WITH
C     ZEROS IN OTHER QUADRANTS
C
      DO 2 KK=1,NN
      K=KK-1
      M1=K*(NN+NTAC)+1
      M2=K*(NN+NTAC)+NN
      DO 2 J=M1,M2
      M3=J-(K*NTAC)
      AX(J)=BY(M3)
      CX(J)=DY(M3)
    2 CONTINUE
      NN=NN+NTAC
      NNN=NNN+NTAC
      N(1)=NN
      N(2)=NNN
      NCPTS=N(1)*N(2)
C
C     TO OBTAIN FOURIER TRANSFORMS
C
      CALL ZERO(BY,NCPTS)
      CALL ZERO(S,NCPTS)
      CALL ARMDFT(N,AX,BY,S)
      CALL ZERO(DY,NCPTS)
      CALL ZERO(S,NCPTS)
      CALL ARMDFT(N,CX,DY,S)
C
C     WE CALCULATE CROSS-POWER AND STORE REAL PART IN X-VECTOR
C     AND THE IMAG. PART IN Y-VECTOR. THUS WE OBTAIN CROSS POWER =X(J)+
C
      J=0
```

```
      DO 80 KK=1,NN
      DO 80 JJ=1,NNN
      J=J+1
      X(J)=AX(J)*CX(J)+BY(J)*DY(J)
      Y(J)=BY(J)*CX(J)-AX(J)*DY(J)
   80 CONTINUE
C
      IF(ISS.NE.0)  GO TO 1
      J=0
      DO 10 KK=1,NN
      DO 10 JJ=1,NNN
      J=J+1
      AX(J)=SQRT(AX(J)**2+BY(J)**2)
      CX(J)=SQRT(CX(J)**2+DY(J)**2)
      AX(J)=AX(J)/CX(J)*1E09.
   10 CONTINUE
    1 CONTINUE
C
C
      RETURN
      END
```

```
      SUBROUTINE  TRANSF(DY,WORK,N)
C     TRANFERS DATA FROM MATRIX TO VECTOR.
      DIMENSION  DY(3000),WORK(26,26)
      K=1
      DO 1 I=1,N
      DO 1 J=1,N
      DY(K)=WORK(I,J)
      K=K+1
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE  CONV 2D(X,F,N4,LF)
C     TWO DIMENSIONAL CONVOLUTION PROGRAM.
      DIMENSION  X(26,26),F(3,3),WORK(3,3)
      ITER=N4-LF+1
      DO 1 I=1,ITER
      DO 1 J=1,ITER
      DO 2 I1=1,LF
      DO 2 J1=1,LF
      I2=(I-1)+I1
      J2=(J-1)+J1
      WORK(I1,J1)=X(I2,J2)
    2 CONTINUE
      SUM=0.0
      DO 3 I1=1,LF
      DO 3 J1=1,LF
      SUM=SUM+F(I1,J1)*WORK(I1,J1)
    3 CONTINUE
      X(I,J)=SUM
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE ZERO(X,NCPTS)
C     SETS THE VECTOR TO ZERO
      REAL X(10)
      DO 1 J=1,NCPTS
      X(J)=0.0
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE  PLOTER(Z,NN)
      DIMENSION  X(26),Y(26),Z(26,26),WORK(1000)
      CALL  PLOTS(WORK(1),4000)
      DO 1 I=1,NN
      DO 2 J=1,NN
      X(J)=4.0+J
      Y(J)=28.0-I
      IF(Y(J).GT.1.0)  GO TO 11
      X(J)=44.0+J
      Y(J)=54.0-I
   11 CONTINUE
      X1=X(J)-0.15
      Y1=Y(J)+0.1
      CALL  SYMBOL(X(J),Y(J),0.05,4,0.0,-1)
      CALL  NUMBER(X1,Y1,0.1,Z(I,J),0.0,3)
    2 CONTINUE
    1 CONTINUE
      CALL  PLOT(40.0,0.0,-3)
      CALL  PLOT(0.0,0.0,999)
      RETURN
      END
```

```
      SUBROUTINE WRIT(AX,NN,NNN)
C     TO PRINT OUT THE RESULTS
      REAL AX(10)
   36 FORMAT(1X,10E13.4)
   45 FORMAT(1H0)
      K=1
      L=0
      DO 1 M=1,NN
      L=L+NNN
      WRITE(6,36) (AX(J),J=K,L)
      WRITE(6,45)
      K=L+1
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE  WRIT2D(X,N)
      REAL   X(20,20)
 30   FORMAT(1X,10E13.4)
 45   FORMAT(1H0)
      DO 1 I=1,N
      WRITE(6,30)  (X(I,J),J=1,N)
      WRITE(6,45)
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CONJG(Y,NCPTS)
C     TAKES THE COMPLEX CONJUGATE OF THE IMAG. PART
      REAL Y(10)
      J=0
      DO 1 J1=1,NCPTS
      J=J+1
      F=Y(J)
      Y(J)=-F
    1 CONTINUE
      RETURN
      END
```